

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

USER MANUAL

FOR

EDR-209 DIGITIZER

3 Channel Model No. 2400-280-3BCF3
6 Channel Model No. 2400-280-6BCF3

February2021

EDM 026 **Issue 4**

Earth Data, Nutsey Lane, Totton,
Southampton, Hampshire, SO40 3NB,
England
Tel: +44 (0) 2380 869922
Fax: +44 (0) 2380 860800
E-mail: sales@earthdata.co.uk

Specification subject to change without notice.

A MEMBER OF THE KENDA GROUP

TABLE OF CONTENTS

1 Introduction 6

1.1 System Architecture 6
1.2 Input Data 6
1.3 GPS Timing 7
1.4 Timing and Accuracy 7
1.5 Data Formats 8
1.6 Phase Locked Loop Data File 10
1.7 Leap Seconds 10
1.8 Seismometer Calibration 11
1.9 Sample Rates 11
1.10 Dual Sample Rates 11
1.11 Dual Sample Rate Limitations 12
1.12 Window Operation 12
1.13 Burst Mode Recording 12

2 Preparing The Unit For Deployment 13

2.1 Read This First 13
2.2 Connecting Up The Unit 13
2.3 Connecting To The Unit 14
2.4 Serial Console 14
2.4.1 Start-Up Scripts 14
2.4.2 Memory 14
2.4.3 Devices 15
2.4.4 Log Files 15
2.5 Telnet 15
2.6 Web Interface 16
2.6.1 Network Configuration 17
2.6.1.1 Network Settings 17
2.6.1.2 Web Access Protection 18
2.6.1.3 Wi-Fi 19
2.6.2 Recorder Configuration 20
2.6.2.1 Channel Summary 20
2.6.2.2 Primary Channels 21
2.6.2.3 Secondary Channels 22
2.6.2.4 Auxiliary Channels 23
2.6.2.5 Time Source Configuration 25
2.6.2.6 Files 29

2.6.2.7 Miscellaneous 33
2.6.2.8 Timers 35
2.6.2.9 Data Streams..... 36
2.6.2.10 SEEDLink 38
2.6.2.11 CAPS..... 39
2.6.2.12 QSCD.....40
22QQ
2.6.3 Status & Utility 41
2.6.3.1 Recorder Status 42
2.6.3.2 Digital Outputs 43
2.6.3.3 Calibration 44
2.6.3.4 Mass Positioning 45
2.6.3.5 WiFi Power 46
2.6.4 Media / Data 47
2.6.4.1 Download Data 47
2.6.4.2 Copy Files. 48
2.6.4.3 Browse 49
2.6.4.4 Format 53
2.6.5 Real-time Data 55
2.6.6 Firmware Update 56
2.7 MODMON 58
2.7.1 Setting The Input Configuration 60
3 Installation And Operating Instructions 61
3.1 Making The External Connections 61
4 Detailed Configuration 63
4.1 Overview 63
4.2 The Configuration File “config.ini” 64
4.2.1 Sections 64
4.2.2 Comments 64
4.2.3 Sample Rates 65
4.2.4 Channel Gain 65
4.2.5 Sample Size 66
4.2.6 Channel Names 66
4.2.7 Channel Usage 66
4.2.8 Channel Formats 67
4.2.8.1 Primary Data Channels 67
4.2.8.2 Secondary Data Channels 68
4.2.8.3 GPS Channel..... 68
4.2.8.4 Auxiliary Channels 68
4.2.9 USB Interface 68

4.2.10 Serial Data Output 69
4.2.11 Serial Baud Rate 69
4.2.12 Ethernet Interface 69
4.2.13 Ethernet Proprietary Data 70
4.2.14 Ethernet Seedlink Data 70
4.2.15 Mseed Master Storage 71
4.2.16 Mseed Secondary Storage 71
4.2.17 Local Volatile Storage 71
4.2.18 Mseed Endian 72
4.2.19 Mseed File Sizes 72
4.2.20 Mseed Header Fields 72
4.2.21 Initial Position 72
4.2.22 GPS Command 73
4.2.23 GPS Power Cycling 73
4.2.24 External GPS 73
4.2.25 GPS Power Saving 73
4.2.26 Digital Outputs 74
4.2.27 Sensor Calibration 75
4.2.28 Sensor Mass Positioning 75
4.2.29 Burst Mode Recording 76
4.2.30 Window Timer 76
4.2.31 Log File Size 77

5 RS232 (RS485) Control and Data Formats 78

5.1 Description 78
5.2 Connection 78
5.3 Transmit Packet Format From The Digitizer 78
5.3.1 Legacy Mode 78
5.3.1.1 MOD Header 79
5.3.1.2 MDE Header 85
5.3.1.3 Data Segment 86
5.3.1.4 Checksum Segment 84
5.3.2 Compressed Mode 85
5.3.2.1 MO2 Header 86
5.3.2.2 Data Segment 87
5.3.2.3 Checksum Segment 87
5.4 Command Strings 97
5.5 Data Retransmission 98
5.6 Channel Selection & Gains 100

5.7 GPS Commands 101
6 Connectors 102
6.1 Output Panel 102
6.1.1 Power Connector (4 way) 102
6.1.2 Ethernet/MODMON RS232 Connector (7 way) 103
6.1.3 GPS Antenna Connector 103
6.1.4 Multi Connector (19 way) 104
6.2 Input Panel 105
6.2.1 Sensor Connectors (10 way) 105
6.2.2 AUX Input Connector (7 way) 106
7 Specification 107
7.1 Input 107
7.2 A to D 107
7.3 Additional Connector Functions 108
7.4 Additional A to D Channels 108
7.5 Communication 108
7.6 Additional Features 108
7.7 Environmental 108
7.8 Power Requirements 109
7.9 Physical 109
8 Accessories 110
8.1 Connecting Cables 110
8.1.1 Sensor Cable 110
8.1.2 Auxiliary Cable 111
8.1.3 Serial Cable 111
8.1.4 Power Connector 112
8.2 Connecting Cables 112
8.3 Connectors 112
9 Upgrade Procedure 113
10 Warranty and Returns 114

1 Introduction

This document applies to the Earth Data ARM based range of digitizers with firmware version V2.xx. For equipment running later software versions, consult factory for validity of this manual. Some older hardware versions may not support all functions described in this document.

The EDR-209 series of digitizers are variable sample rate 24 bit digitizers using unique Delta-Sigma converter technology developed by Earth Data. The Delta-Sigma converter has an input range of ±8 volts and gives a signal/noise figure of better than –140dB at 100 samples per second. Two alternate gain settings are available giving input ranges of ±0.8 volts and ±20 volts.

All internal units are specified for operation over a wide temperature range. The unit is intended for use with an external GPS antenna that disciplines the internal time base.

The unit is supplied as a three or six channel unit. The internal digitiser can be monitored in near real time via the Ethernet, a RS232/RS485 link to an external PC or laptop. All software and firmware can be remotely updated, avoiding costly return to factory or dismantling.

An internal compact flash allows about 60 days of continuous recording at 100 samples/second. When the disk is full the oldest data is normally replaced with new data and at any time the previously recorded data is available by downloading via an Ethernet connection. Ethernet downloads is undertaken without interrupting the acquisition process.

All recorder parameters are set by a configuration file, which is copied to internal flash before use. This is written using any standard text editor on a separate PC or directly on the unit. Since this file fully characterises the machine, the recorder merely implements the configuration contained in this file.

1.1 System Architecture

The digitizers are controlled with an ARM-RISC based microprocessor running the ARM based version of Linux. The unit is constructed using a set of 5 PCBs housed in a ruggedised weather-sealed box. An internally mounted GPS with an external antenna synchronises the unit to UTC time with a digital phase locked loop maintaining close time whenever the GPS unit loses satellite lock.

1.2 Input Data

The units convert analogue signals with a dynamic range of up to 150 dB at various sample rates between 1 and 1000 samples/sec. The lower sample rates produce higher dynamic range since the bandwidth is restricted. All channels must have the same sample rate when multiplexed legacy mode data is used. There is no such restriction with un-multiplexed

compressed data mode is used. The number of bits representing each sample transmitted over RS232/485 can be set to 24 or 32 bits. 32 bits can be used to take advantage of the high dynamic range at low sample rates. At higher sample rates the noise floor is such that 24 bits is sufficient.

At 24 bits the minimum resolvable range is normally ±8 volts (1µV / bit) but can be increased to 100nV/bit by increasing the gain of the input stage by 20 dB. There is a very low gain range available allowing an input range of ±20 volts (2.5 µV / bit).

In legacy mode the data is transmitted in one second packets once a second. The packet is made up of signal data preceded by a header containing information about the digitizer operating mode and various other status information. A checksum block is included at the end of the signal data. The GPS message from the GPS is included in this header when lock has been achieved for the first time. An enhanced header can be enabled from the CONTROL page, which gives more information when six channels are selected or alternative GPS units are fitted although this may require a higher RS232 baud rate.

1.3 GPS Timing

The digitizer uses a Trimble iQ GPS receiver, which provides synchronisation to UTC time. The GPS receiver is integral to the digitizer and the unit operates simply by connecting an external aerial to the digitizer using the cable supplied. The standard GPS antenna is a 3V "bullet" type. The GPS cable is normally 15 metres in length, but can be supplied in shorter lengths if required. For GPS cables longer than 15 metres other options are available (consult Earth Data for details). The GPS has an internal battery to preserve unit position data when the unit is powered off. An initial deployment where the unit has moved position may take up to 30 minutes to gain lock, subsequent locks are normally gained within a four minutes.

When GPS lock is achieved for the first time the packets are locked to the GPS one pulse per second. A time shear will result possibly producing several incorrect packets. Subsequent loss of the GPS signal will not produce any further shears, since the timing now free wheels until the GPS one-second pulses are re-established. The error between the current GPS one-second pulse (if present) and the internal time is reported in each packet.

1.4 Timing and Accuracy

All timing is referenced to the internal GPS. When a recorder is first switched on all time reference is against the internal PC clock which in reality gives accuracy to no better than seconds. When the GPS first gains time lock the internal digitally controlled Phase Locked Loop (PLL) Oscillator is aligned to GPS time giving accuracy to microseconds. This internal oscillator now tracks the GPS time and is constantly pulled in time to match the GPS. While the internal oscillator is tracking the GPS the voltage pattern used to control the oscillator is noted. This average voltage is then used to maintain an idealised time source whenever the GPS loses lock. When the GPS regains lock the internal oscillator is

once more pulled to match the GPS. The discrepancy between GPS time and the internal time is stored in microseconds on a second by second basis.

Under conditions of good GPS reception the unit provides an accuracy of ±10 microseconds with respect to GMT.

GPS/phase information and GPS status is recorded in one of four formats which is selectable via the configuration file.
The object is to record accurate timing information whilst conserving disk space. Irrespective of which option is selected at least one complete GPS message is saved in each day directory.

1.5 Data Formats

Data output is available in a number of forms.

The original interface (Earth Data, Legacy One Second packets) sends data in one-second packets together with a header holding GPS time information, auxiliary channel data, temperatures etc. The data channels are multiplex together with each channel containing exactly the number of samples for one second. This data is available over a serial RS232/RS485 interface and is compatible with earlier Earth Data Digitizer units.

A highly compressed loss-less interface is available which compresses sample data by storing successive differences between samples reducing sample size from 24 bits down to typically 6-10 bits on average. Channel data is built up in variable length packets on a channel-by-channel basis and are transmitted together with time information when a packet is complete. This results in efficient bandwidth usage but means that sample data is not available on a periodic basis. Channel data that compresses better is transmitted less often. This data is available over the RS232/RS485 or via Ethernet and TCP/IP.

A third interface is the SEEDLINK protocol as produced and maintained by GFZ at Potsdam University in Berlin. The digitizer with optional inbuilt disk storage is capable of supporting a SEEDLINK server transmitting data over Ethernet TCP/IP. Refer to http://ds.iris.edu/ds/nodes/dmc/services/seedlink/ and https://www.seiscomp3.org/wiki/doc/applications/seedlink for further information on SEEDLINK.

Finally data can be transmitted as a simple data series in an ASCII format.

The default output data is Earth Data one-second packet format sent over the RS232 serial interface. This is the Earth Data multiplexed packet format (legacy mode data) and is compatible with older issues of digitizers. This may be sent over an Ethernet link using TCP/IP or for longer serial links over a RS485 link.
The RS232/RS485 interface is bi-directional allowing the unit to be configured to the users requirements. This same interface can be used for storing the settings in EEPROM within the digitizer and downloading later issues of the operating firmware within the digitizer.

Either of the above formats is available over an Ethernet network. Data is pushed out through a logical socket to a host or data can be retrieved via FTP or RSYNC. Configuration of the unit can be via a standard Web Browser, a "telnet" session or via FTP and a configuration file.

There are four auxiliary analogue inputs that are sampled at about one Hertz. There are two digital outputs for controlling seismometer centring etc.

The 2400-280-3BCF3 digitizer is a 3-channel version having "MIL–26482" bayonet type connectors. The 2400-280-6BCF3 is the 6 channel version.

The unit is powered from a D.C. source of 12 to 28 volts and is protected against reverse polarity connection and transient over-voltage. Continuous over-voltage will cause the internal fuse to blow without any further damage. Power consumption is below 1.0 Watts for a three channel unit and less than 1.5 Watts for a six channel unit.

Four extra low-resolution inputs are provided for monitoring slowly changing parameters. These are sampled by a conventional 16 bit ADC at 1 sample per second.

Compressed mode data is built up individually for each channel. Creating de-multiplexed data at source allows it to be compressed relatively simply by storing the difference of consecutive samples instead of the absolute value of the sample. Each packed so built up contains an absolute start sample value and the time of that sample, followed by a set of differences for the following samples.

The signal input connections to the high-resolution channels are differential, each input presenting an input impedance of 1M ohm to ground. When operating with a passive sensor with floating sense coils, the signals will be truly differential and will see an input resistance of 2M ohm. If the output from the sensor is single ended, then the signal can be applied to either of the input poles with the sensor ground connected to the other pole. Refer to section 3.1 for more information on single ended connection. The input resistance for single ended connection is 1M ohm.

The additional low-resolution analogue inputs are sampled at 1 sample per second irrespective of the selected sample rates for the main channels. The converted values appears once a second in the packet header.

The digitizer is supplied with a Windows utility MODMON32 for verifying correct operation of the digitizer. This provides a near real-time display of the time series for each channel, an FFT for each channel and GPS status. Header information is also displayed in an easily read manner. MODMON32 can be used for setting the required operating parameters and saving the settings to the digitizer EEPROM memory.

MODMON32 will operate over the Ethernet or serial RS232/485 connection.

When the digitizer is connected to the Ethernet any Web browser can configure the unit. The digitizer hosts a Web Server that allows remote configuration and status display. A low-resolution version of input data can be displayed.

1.6 Phase Locked Loop Data File

The phase locked loop (pll) data file combines GPS lock data and internal time tracking data to provide a timing quality indication. From this file at any time GPS lock can be determined and where GPS does not have lock

The pll data file has the format
date time pll_age pll_data

date The date in the format ddmmyyyy that this entry refers to.

time The time of day in the format hhmmss that this entry refers to.

pll_age When the GPS loses lock the phase data cannot be determined so the last valid phase data is logged together with the age of the phase data in seconds. This entry denotes one of three conditions as indicated by its value.
An age of –1 denotes that the GPS has not had lock since power up.
An age of 0 denotes the GPS has lock and the phase date and time is current.
An age greater than 0 denotes how long in seconds the GPS has been out of lock for.

pll_data This is a signed integer in microseconds. A positive value indicates that the recorder is reporting the data occurring earlier than actual time (the recorders internal time is running slow). A negative value indicates the recorder is reporting data that has occurred later than actual time (the recorders internal time is running fast).

The GPS status file has the format
date time lock_status antenna_status

lock_status 0 for lock, 1 for no lock

antenna_status 0 for no problems, 1 for open circuit, 2 for short circuit.

Pll phase data is stored in a single file for each day with the name format the same convention as the data files but with hhmmss set to zero. i.e.
/data/day_number/yymmdd000000.pll

Pll status (lock and antenna status) is stored in a similar file with a **gst** extension.

Selecting a pll format (gps2,gps3,gps4) suppresses the gps channel to conserve disk space. A single gps message will be recorded each day to provide positional verification.

1.7 Leap Seconds

Checking for sequential time codes from the GPS caters for time slips and leap seconds. The digitiser continuously monitors the GPS time message and compares the time in the

GPS message against the tracked time of the digitiser phase locked loop circuitry. Wherever a discrepancy occurs the current data file is terminated at the time slip and a new data file is started.

This detection provides a solution for leap seconds by providing a data file that is either one second long or one second short catering for the addition or removal of the leap second.

A leap second can only be detected if the GPS has lock at the time the leap second occurs. The digitiser will only know about leap seconds when GPS lock is re-established. When the GPS does not have good lock at the leap second post processing will be required to recover the time data between the actual leap second and the time it is actually detected (GPS regains lock). If GPS has lock at the instant of the leap second the current data file will be terminated at the shear and will be one second long or short.

1.8 Seismometer Calibration

Simple seismometer calibration pulses are available in version 2.02. There are two logic outputs (1 CMOS driver, 1 open collector) and an analogue output available for seismometer calibration. Refer to sections 4.2.27, **Error! Reference source not found.**and **Error! Reference source not found.** for the format of the inifile entries. The calibration timings have an onset accuracy of ±1 millisecond.

1.9 Sample Rates

The unit will sample data at rates of between and including 1 sample/second to 3000 samples/second. The available rates in samples/second are:- 1,2,4,5,10,20,25,40,50,75,100,120,125,150,200,250,300,375,500,600,750,1000, 3000.

1.10 Dual Sample Rates

The unit can sample each seismometer at up to two data rates. Channels 0 to 5 provide the primary (higher) sample rate while channels 6 to 11 provide a secondary (lower) sample rate. The channels are assigned like :-

Actual Channel	Primary Channel	Secondary Channel
0	0	6
1	1	7
2	2	8
3	3	9
4	4	10
5	5	11

1.11 Dual Sample Rate Limitations

There are no limitations applied to the secondary sample rate. Any primary sample rate can have any secondary rate.

1.12 Window Operation

Not currently available.

1.13 Burst Mode Recording

Burst mode recording allows high sample rate data to be collected for short periods of time whilst recording continuously at a lower rate. This is a special mode of dual rate recording. The secondary data channels (channels 6 to 11) record data continuously while the primary channels (channels 0 to 5) record data normally at a higher rate but only for periods specified by an inifile entry. Each channel is configured as normal for dual sample rate recording together with an additional entry in the inifile specifying that the primary channels operate in burst mode recording. Using the "burst" entry (section 4.2.29) recording on the primary channels is switched on and off either as a one off operation or repeated at intervals.

2 Preparing The Unit For Deployment

2.1 Read This First

Certain functions at the current software issue (V3.20) are not currently implemented but described in this manual and will be included in a future release.
At the current software issue (V3.20) the following items are not supported and will be added to a future release.

Low power timing windows is not supported.
Trigger logging is not supported
GPS initialisation strings are not supported
An external GPS is not supported
Seismometer calibration not supported

2.2 Connecting Up The Unit

Identify the pin numbering of the connectors by referring to section 6

A nominal 12-volt supply should be connected to the unit. Power consumption should be less than 0.9 Watt for a three channel unit and less than 1.2 Watts for a six channel unit.

Powering the unit causes the output panel LEDs to flash. There is a flash sequence through which the LEDs sequence during its power up. The LEDs flash at 2Hz during the power up process. In the following table "X" denotes a led flashing at 2Hz and "O" denotes a LED that is off. The complete boot process takes about 70 seconds in total. After which time the recorder is operational and you may log into the unit.

ACQ	GPS	PC	Time (secs)	Action
O	O	O	-	Power supplies are stabilising
X	X	X	4	All LEDs flash three times.
X	X	O	6	Primary bootstrap
X	O	X	23	Unpacking and starting Linux
O	O	X	66	Linux Operational
O	X	O	67	Login available
X	O	O	68	Recorder starting
LEDs indicate recorder operation from this point				

Onwards from the point that the acquisition is running Linux has reached its normal run level and the unit is available via the Ethernet and may be communicated with via a "telnet" session or via a web browser.

A word of caution here, the unit can be slow responding during the initial minute of recorder operation especially if a USB device is connected and/or “Seedlink” is starting. The device mount processes can consume a large amount of processing time. Once the mounts have completed and especially after “Seedlink” has started the unit will become responsive again.

2.3 Connecting To The Unit

A functional check of the digitizer can be quickly carried out via the Ethernet. Connect the digitizer to a laptop or network using the Ethernet interface. The digitizer is a peer and needs a standard Ethernet cable co connect to a network hub. When connecting directly to a laptop a peer-to-peer swap over cable (these are normally brightly coloured) will be required.
Initially the units are supplied with one of a preset set of IP addresses.
192.168.123.53
192.168.123.156
192.168.123.190 to 192.168.123.205

2.4 Serial Console

In addition to the Ethernet port and serial MODMON port there is a serial port which is the unit console. This port is not used in normal operation but is available for user use and may be particularly useful when initially deploying units. This can be thought of as a debug port, which enables initial connection to the unit. This port is in fact the Linux console. A description of the Linux system is well beyond the scope of this document, however a brief description of the system is suitable and will allow experienced Linux users to understand a little about how the system is organised. In particular to aid any user implemented scripts or procedures. . Its parameters are fixed at 115200,N,8,1.

2.4.1 Start-Up Scripts

The script **/mnt/mtda/users/startup** is the first script run after default Linux boot-up and configuration which invokes **/mnt/mtda/users/recorder**. All other processes are then loaded from the recorder process “recproc”. User supplied scripts may be invoked from either of these scripts.

2.4.2 Memory

The following three devices are actually ram devices and as such use main memory space. You should be very careful if you use these devices as a storage medium and be aware the contents will be lost when the unit is powered off.
proc mounted on **/proc**
sysfs mounted on **/sys**
tmpfs mounted on **/dev**

If you use your own customized scripts or processes as a guide they should not occupy more than 10Mb of memory.

2.4.3 Devices

Device **/dev/mtdblock1** mounted on **/mnt/mtda** is one of the onboard flash devices containing scripts and executables that configure and run the device as a digitizer.

Device **/dev/mtdblock2** mounted on **/mnt/mtdb** is a second onboard flash that contains the configuration and executables for "seedlink".

Device **/dev/sda1** mounted on **/mnt/data/usba** is a hotplug USB device and is used to record digitizer data.

Device **/dev/hda1** on **/mnt/data/hda** is an onboard compact flash that is used as the data store for "seedlink"

2.4.4 Log Files

The Linux operating system maintains a boot log. You can access this by entering **dmesg.**

The recorder system maintains its own log in the file **/var/log/rlog**

Linux maintains its own event log in the **file /var/log/messages** or **messages.0**

2.5 Telnet

The unit supports multiple "telnet" sessions. From a "telnet" console enter (say) "open 192.168.123.53"

From the digitizer prompt enter user as "root" and password as "root"
You now have a console session with the digitizer.

A note about "telnet" from a Windows session. The "telnet" session must have the UNIX style of line delimiting, that is to say a line is terminated with a "line feed" and not the Windows way of "carriage return, line feed". You may need to configure "telnet" with the "unset crlf" directive before you can login to the recorder.

2.6 Web Interface

Open a web browser and connect to the unit IP address. The unit runs a Web server that allows its configuration to be changed, its status to be viewed, and a quick view of incoming data. The web interface configures the unit by making changes to the units configuration file **"/mnt/mtda/users/config.ini"**. If as a user you prefer you can edit the configuration file directly with your favourite editor. Changes to a page are implemented with the update button

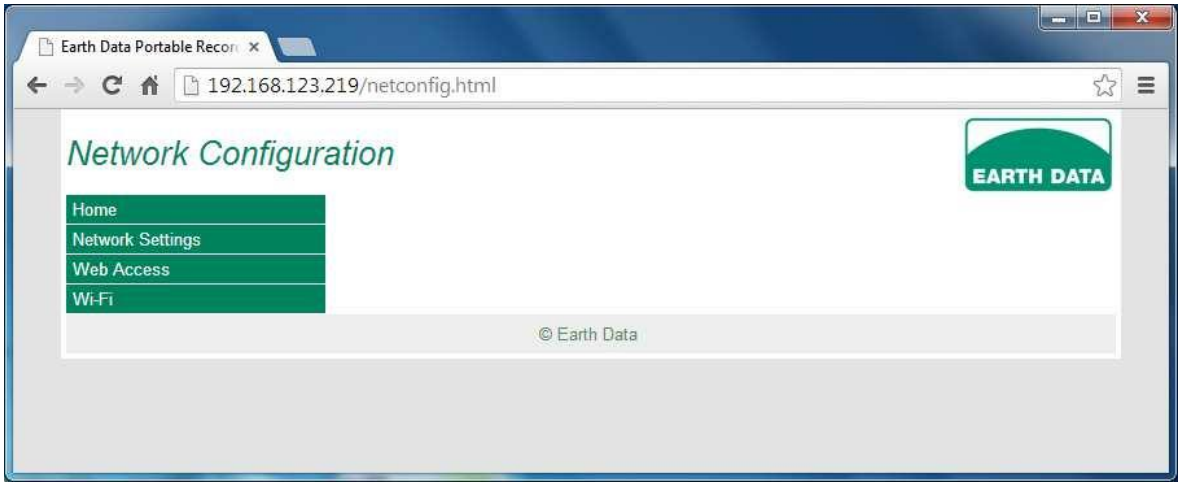


The web server consists of six main sections.

Network Configuration	2.6.1
Recorder Configuration	2.6.2
Status/Utility	2.6.3
Media/Data	2.6.4
Realtime Data	2.6.5

2.6.1 Network Configuration

Select Network Configuration from the main page to change or view the recorder's network configuration.

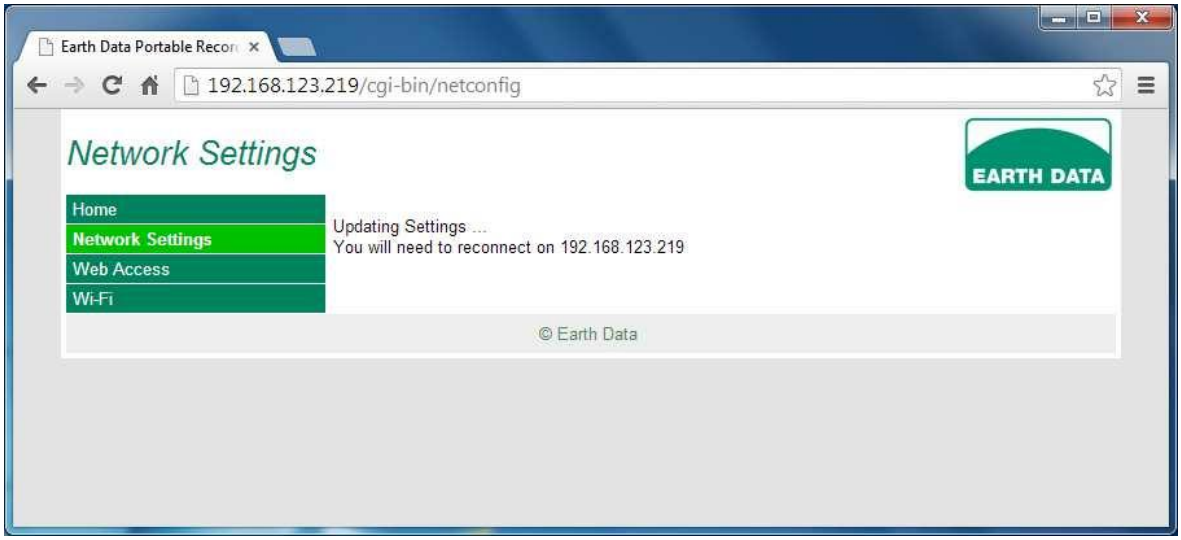


2.6.1.1 Network Settings

Select 'Network Settings' and the current IP configuration can be displayed and / or changed.

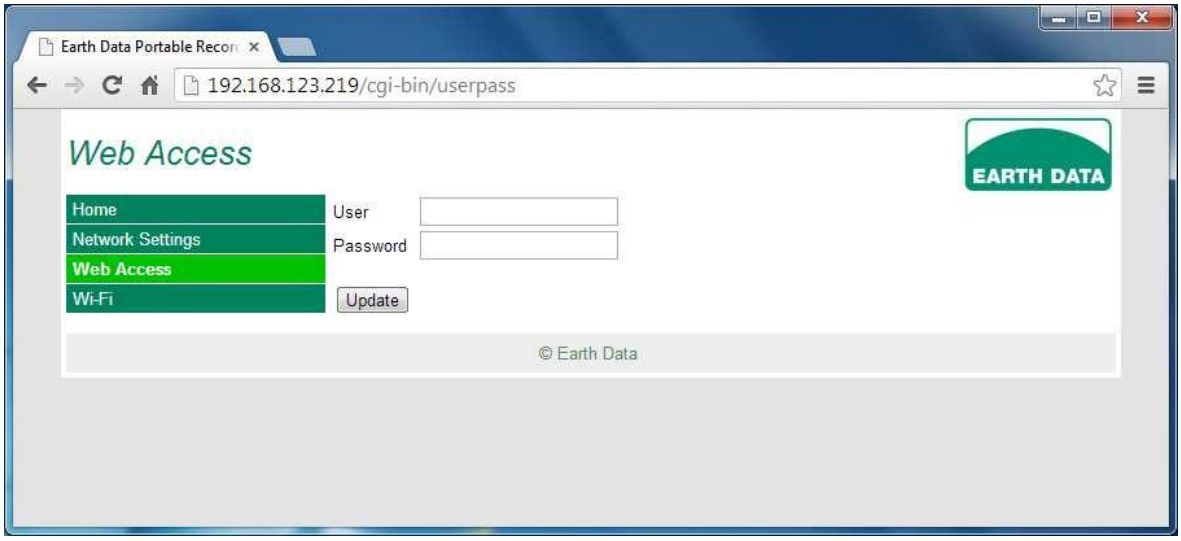


The changes are reflected instantly and you are informed to reconnect on the new address.



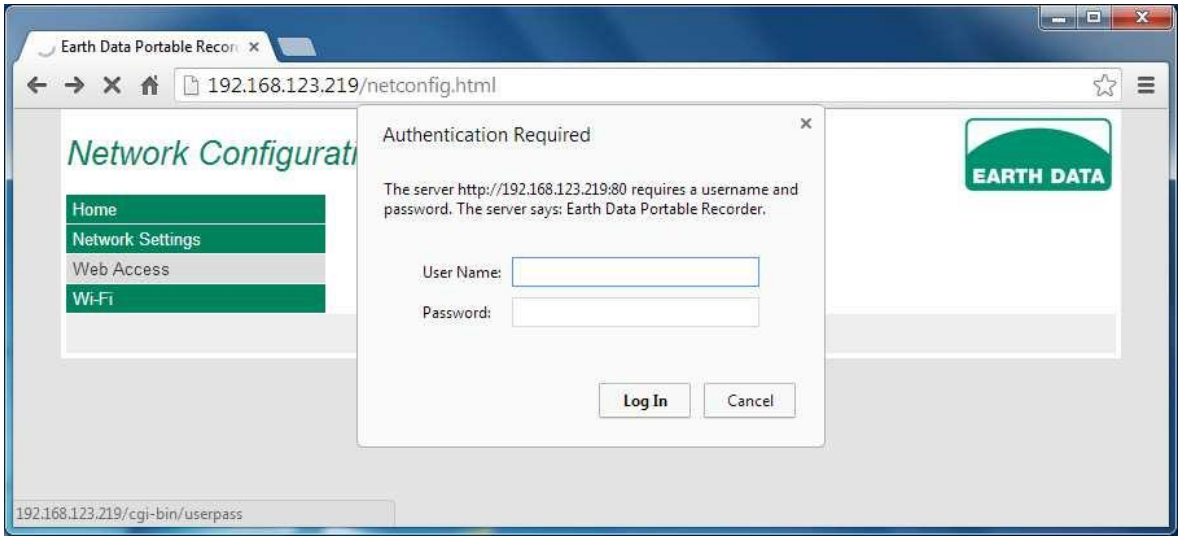
2.6.1.2 Web Access Protection

By selecting 'Web Access' you can password protect web access (which by default is anonymous). This user and password is solely for gaining access to the browser services.



To permit anonymous access, simply leave the User and Password entries for 'Web Access' blank.

Having entered a user and password clients will be prompted to gain access.



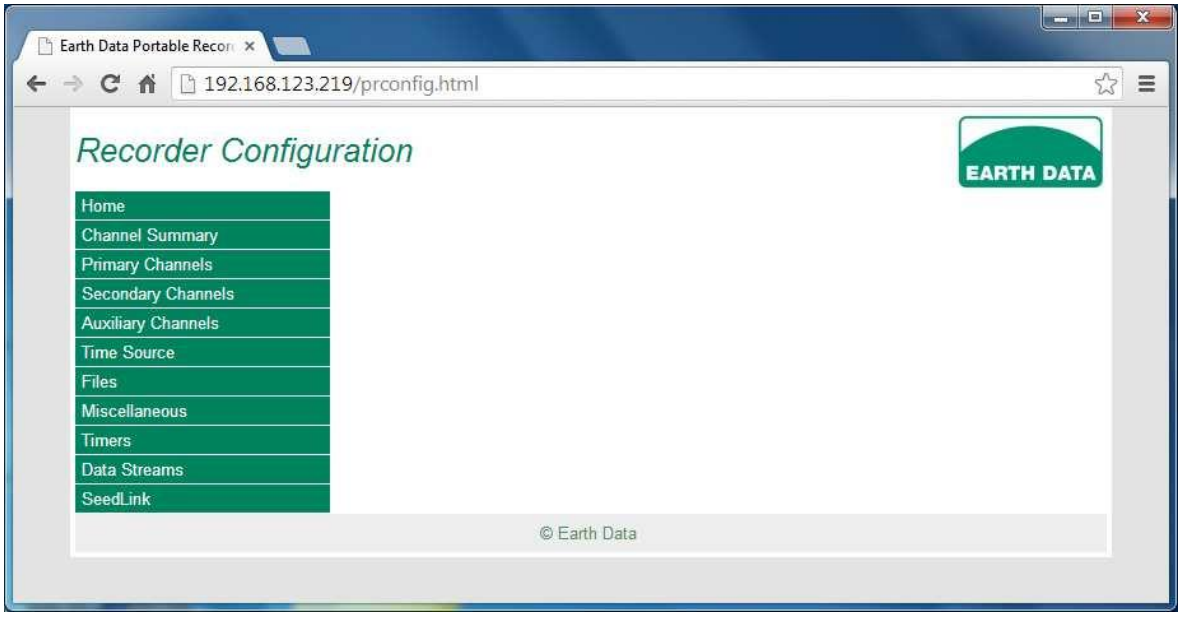
2.6.1.3 Wi-Fi

In order to encrypt the Wi-Fi network connection to restrict access, an encryption key can be set. This will be required when connecting to the recorder's wireless network.



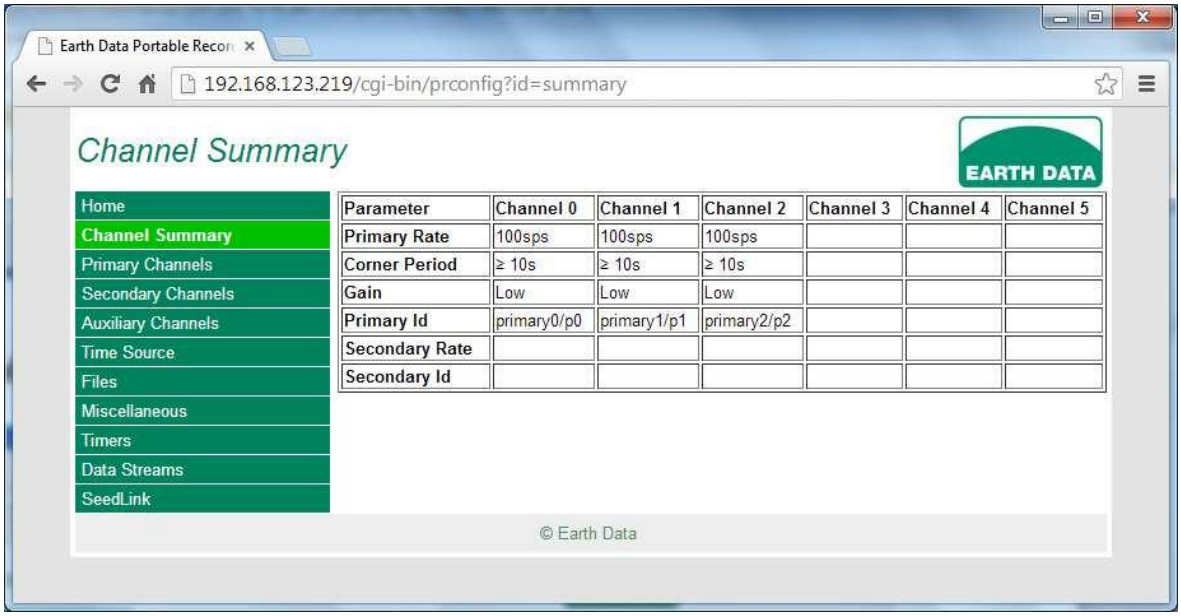
2.6.2 Recorder Configuration

Select Recorder Configuration from the main page to view / edit the current configuration and view the current status.



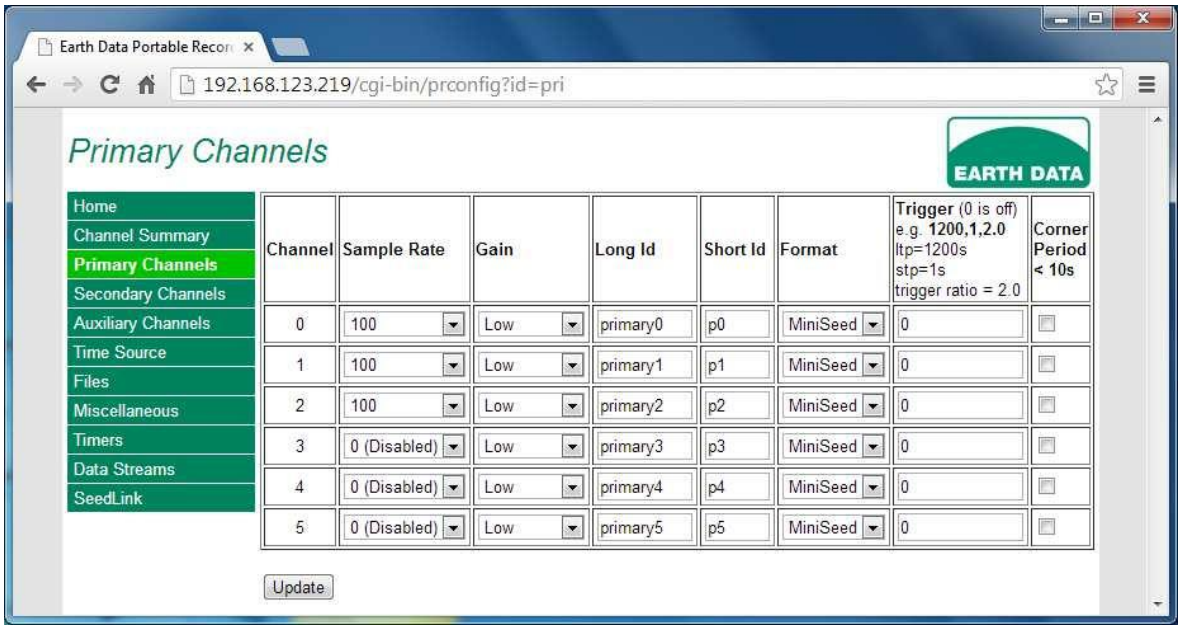
2.6.2.1 Channel Summary

This is a convenient summary page of the primary channel settings.



2.6.2.2 Primary Channels

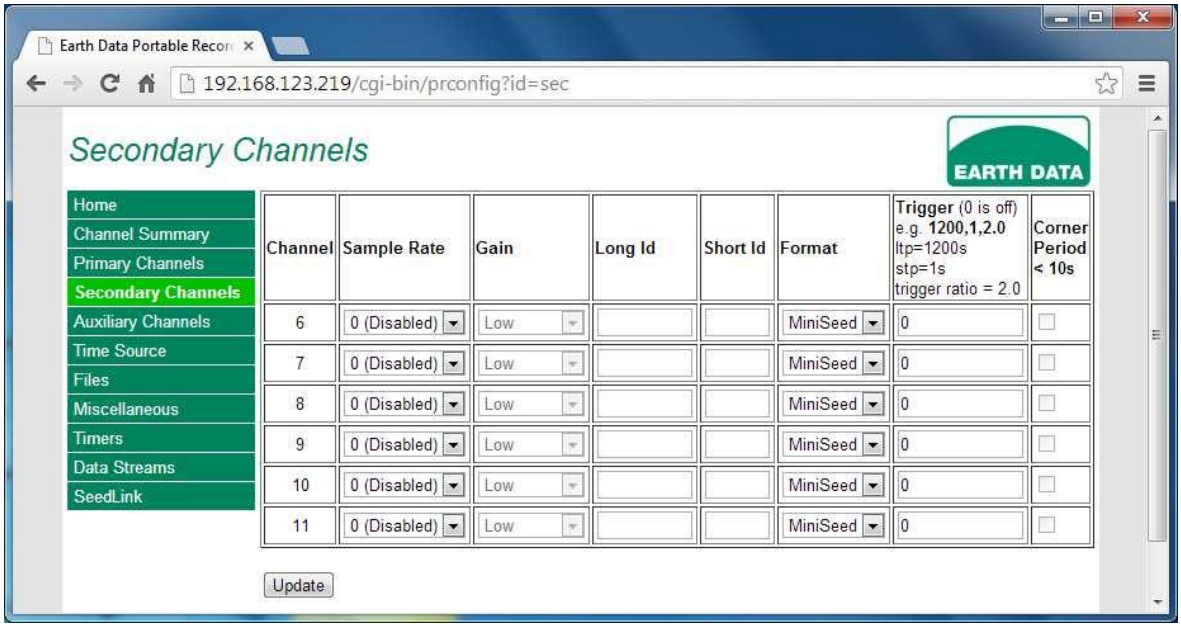
Select 'Primary Channels' to configure the primary channels.



Parameter	Description
Long Id	This entry defines the channel name associated with the relevant primary channel. The above example will generate data files like:- */data/113/edyymmddhhmmss.***pri0**
Short Id	This entry sets the channel identifier in the fixed data header. Only the first three characters are used. If omitted the first three characters from Long Id are used.
Gain	This entry details the gain setting applied to the channel. Four options are available: Low (the default), High, Very High and Very Low.
File Format	This entry details how files are recorded for a given channel. Three options are available: Mini Seed (the default), ASCII and Legacy.
Corner Period	The corner period is for the seismometer attached to a channel and is used by SEEDLink to name the data stream.
Trigger	Not implemented.

2.6.2.3 Secondary Channels

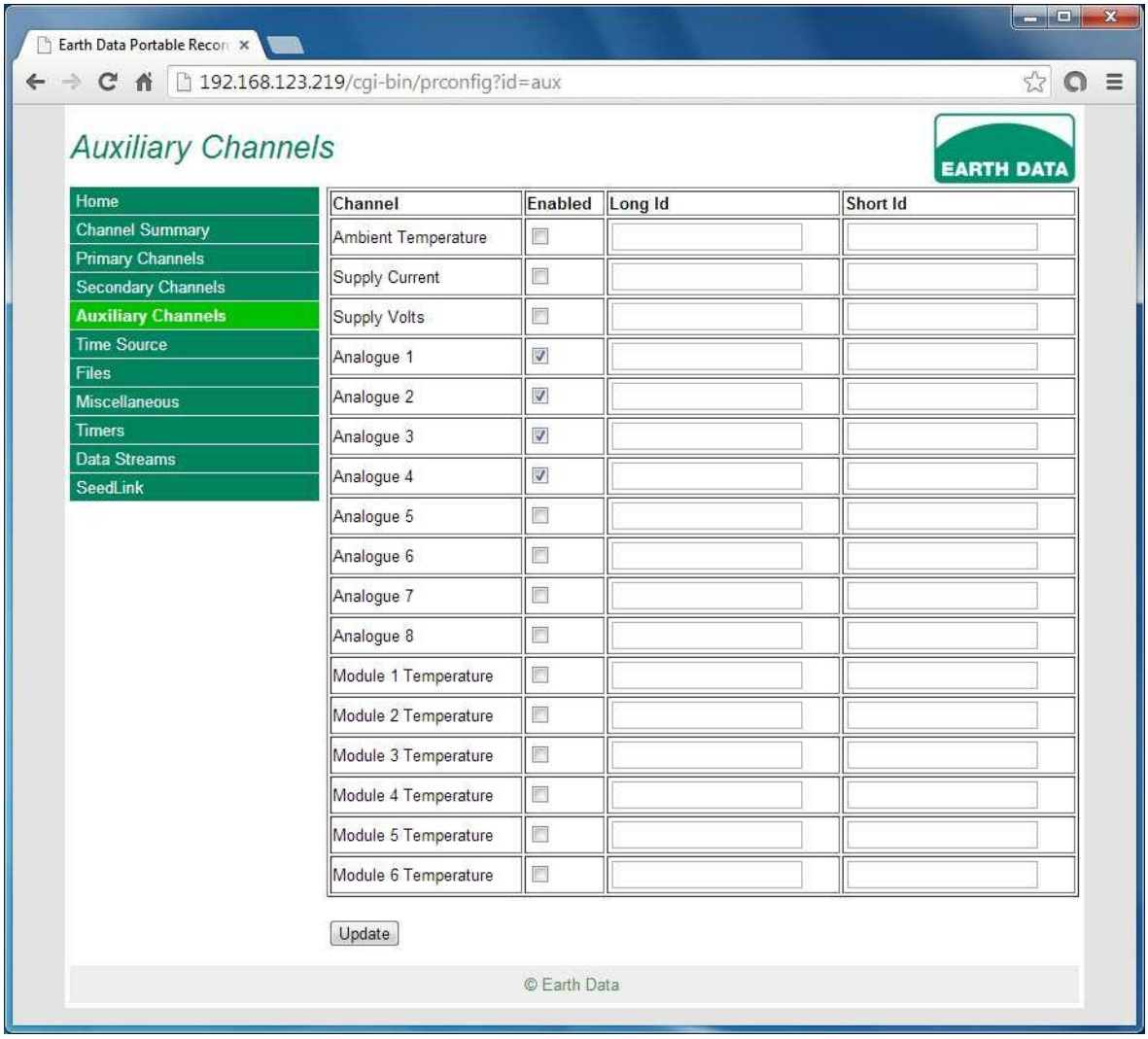
Select 'Secondary Channels' to configure the secondary channels.



Parameter	Description
Sample Rate	This entry sets the secondary sample rate in samples/sec for the channel. A value of zero switches the secondary channel off.
Primary Gain	This entry details the gain setting applied to the primary channel (the same setting is applied to the secondary channel). Four options are available: Low (the default), High, Very High and Very Low
File Format	This entry details how files are recorded for a given channel. Three options are available: Mini Seed (the default), ASCII and Legacy.
Trigger	Not implemented.

2.6.2.4 Auxiliary Channels

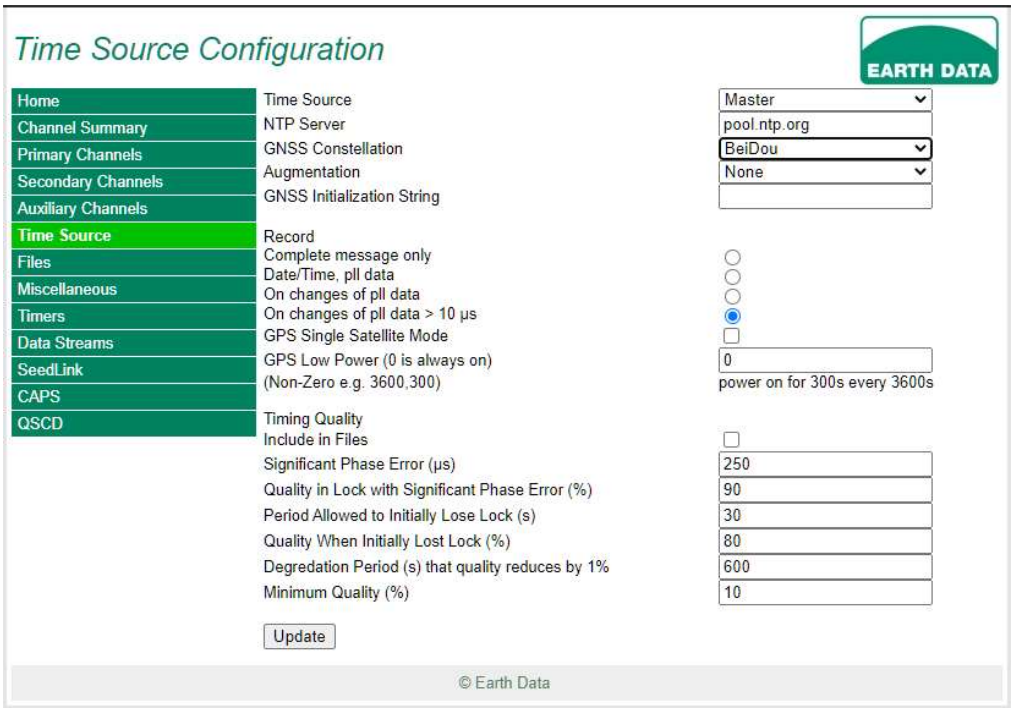
Select ‘Auxiliary Channels’ to configure a given auxiliary channel.



The available Auxiliary Channels are:

Auxiliary Channel	**Description**
Ambient Temp	Ambient temperature in °C
Supply Current	Supply Current (unit load in mA)
Supply Volts	Battery Volts, input supply voltage in mV
Analogue 1	External user input 1
Analogue 2	External user input 2
Analogue 3	External user input 3
Analogue 4	External user input 4
Analogue 5	External user input 5
Analogue 6	External user input 6
Analogue 7	External user input 7
Module Temp 1	Module temperature in °C
Module Temp 2	Module temperature in °C
Module Temp 3	Module temperature in °C
Module Temp 4	Module temperature in °C
Module Temp 5	Module temperature in °C
Module Temp 6	Module temperature in °C
Parameter	**Description**
Enabled	Tick this check box to enable the relevant Auxiliary Channel
Long Id	This entry defines the channel name associated with the relevant auxiliary channel. The above example will generate data files like:- */data/113/edyymmddhhmmss.***ambient Temperature**
Short Id	This entry sets the channel identifier in the fixed data header. Only the first three characters are used. If omitted the first three characters from Long Id are used.

2.6.2.5 Time Source Configuration



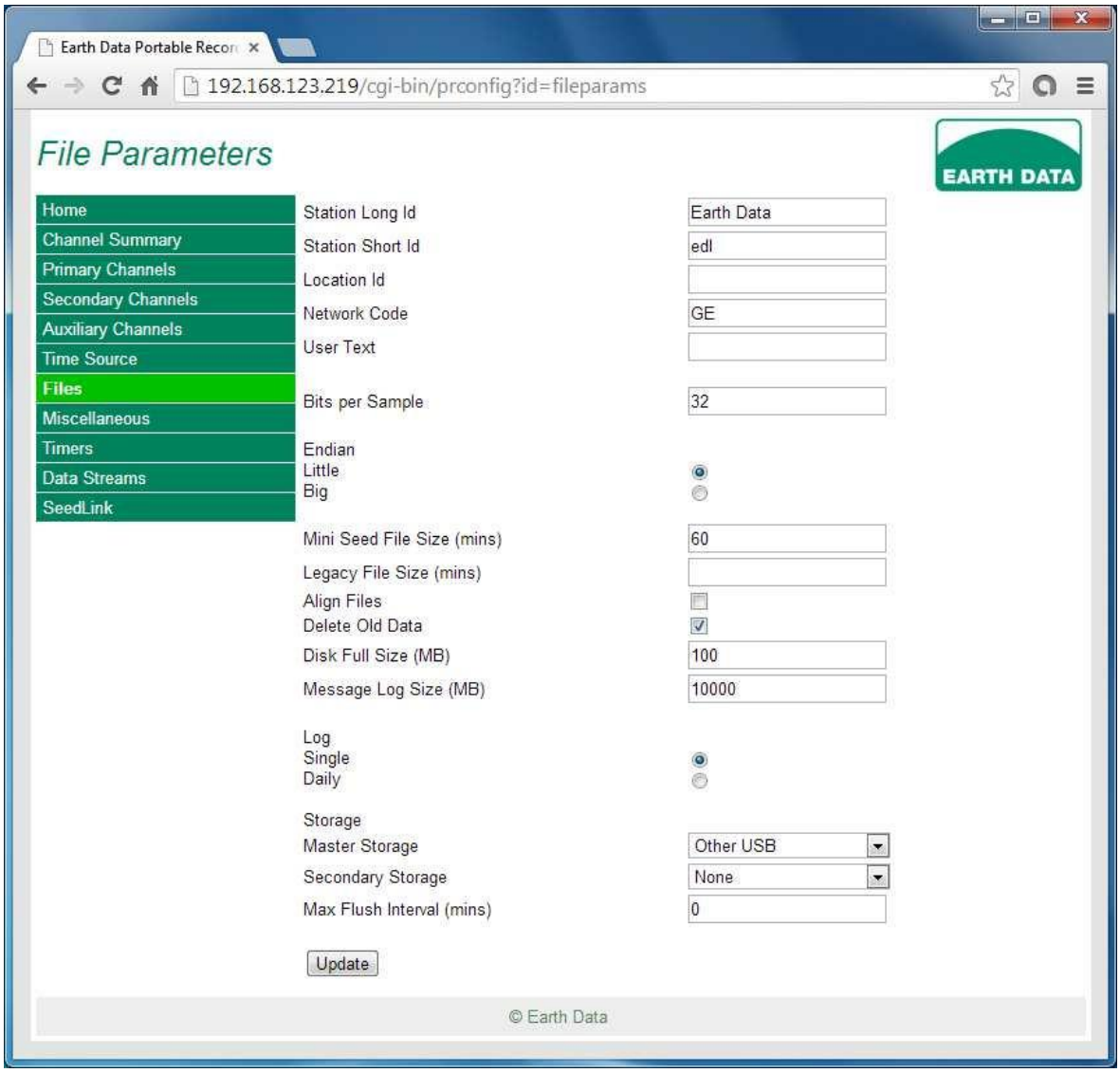
Parameter	Description
Time Source	Valid settings are: **Master:** Unit's GPS is to be master time source. **Slave:** Unit's time is sent from connected master unit. **None:** Unit's GPS is to be disabled – in which case the time source will try and locate a valid NTP server.
NTP Server	IP address of an NTP server to be used in the event that GPS reception is inhibited.
GNSS Constellation	Changes the satellite constellation used by the GNSS receiver
Augmentation	Change the augmentation used by the GNSS receiver

Parameter	Description
GNSS Initialization String	When present this entry will send an ASCII data stream to the GPS during its initialization phase. The ASCII string can be any ASCII printable code up to 62 bytes in length, but must not contain spaces. The full description for this option falls outside the scope of this document, and the Lassen Trimble iQ manual operational manual must be consulted. There is one specific use however for this entry. If a GPS knows its approximate position prior to first use the time to the first lock may be reduced considerably. The format of this type of string is: **>SIP+50+001+0005<**. This is the Set Initial Position command, which in this case sets the latitude to +50 (‘+50’ represents 50° North, ‘-50‘ would represent 50° South), sets the longitude to +001 (‘+001’ represents 1° East, ‘-001’ would represent 1° West) and sets the altitude to +50m (‘+0005’ represents +0005 x 10m above sea level).

Parameter	Description
Record	Four choices are available: 'Complete message only' Original GPS format (mixed ASCII and binary) Records the complete GPS message, no **pll data,** no **status data** 'Date/Time, pll data' Records **date**, **time**, **age** of **pll data**, **pll data** once per second. Records GPS **lock** and **antenna** status once per second. 'On changes of pll data' Records whenever **pll age** or **pll data** changes. Records GPS lock and antenna status whenever it changes. 'On changes of pll data > 10 micro s' Records whenever **pll data** changes and when lock is resumed but is suppressed when the **pll data** is within 10µs of GPS. Records GPS lock and antenna status whenever it changes.
Single Satellite Mode	Tick this entry to set the GPS to its single satellite mode, which may produce a faster initial lock under poor satellite conditions. The full description for this option falls outside the scope of this document and the Trimble IQ operational manual must be consulted.
Low Power	This entry enables power cycling of the GPS unit. The two comma separated arguments are 'cycle time' and 'on time' respectively. Both 'cycle time' and 'on time' are specified in seconds. So for example setting '3600,300' powers the GPS on for 5 minutes (300 seconds – 'on time') once every hour (3600 seconds – 'cycle time'). The default (0) when not specified is to have the GPS always powered. When in power cycle mode the on period cannot be less than 5 minutes (300 seconds) and more than 16.65 minutes (999 seconds). The off period cannot be less than 5 minutes and more than 8 hours (28800 seconds).

Parameter	Description
Timing Quality	Include in files: Selecting this option will create a Blockette1000 in with the Mini-SEED data files. This contains the *timing quality* as a value between 0 and 100%. Significant Phase Error in microseconds. The unit has GPS lock but having previously lost lock or initially acquiring lock is not yet aligned with GPS time. The *timing quality* is degraded to the following value.
Quality in Lock with Significant Phase Error (%).	A significant phase error degrades the *timing quality* to this value.
Period Allowed to Initially Lose Lock (seconds).	When GPS lock has been lost for this period the *timing quality* is degraded to the following value.
Quality When Initially Lost Lock (%).	Having lost lock initially the *timing quality* is degraded to this value.
Degradation Period (seconds) that quality reduces by 1%.	Having lost lock the *timing quality* reduces by 1% for each degradation period that passes until the minimum quality value is reached.
Minimum Quality (%).	Having gained a good GPS lock at some time in the past this is the minimum value that *timing quality* can degrade to.

2.6.2.6 Files



Parameter	Description
Station Long Id	This entry defines the leading characters of data file names. The above example will generate data files like: */data/113/edhmmss.channelname*
Station Short Id	This entry defines the station identifier code in the fixed data header section of a mini-seed file. Only the first five characters are used. If omitted the first five characters of 'Station Long Id' are used.

Parameter	Description
Location Id	This entry defines the location identifier code in the fixed data header section of a mini seed file. Only the first two characters are used.
Network Code	This entry defines the location identifier code in the fixed data header section of a mini seed file. Only the first two characters are used.
User Text	This entry adds a line of text to the log file. This can be any user text up to 17 characters and appears in the log file as "Station: <xyz>". This can be used for any purpose. In addition when the Earth Data Legacy file format is used this text is written to the user text field in the header.
Bits per Sample	This entry specifies the size that samples are saved internally. A full sample is 32 bits. Reducing the size saves on storage requirements.
Endian	This entry specifies the endian order of data in a mini seed file. The default is little endian which is the Intel (PC) word ordering. Specifying endian=big swaps the byte order in words to the endian system used by Motorola / Sun etc.
Mini Seed File Size (mins)	This entry sets the size of the mini seed files. This parameter specifies the data content in minutes. The default when not specified is a mini seed file containing 10 minutes of data. There is a recommended limit on this size that is dependent upon the sample rate of the fastest channel. So for example if there is one or six channels running at 500s/s the maximum file size is sixty minutes. The recommended list is shown in the table below*. A file can never exceed one day (1440 minutes)
Legacy File Size (mins)	This entry sets the size of legacy format files. This parameter specifies the data content in minutes. The default when not specified is a file containing 10 minutes of data. The same file size limits are imposed as for mini seed files. The full list is shown in the table below*.

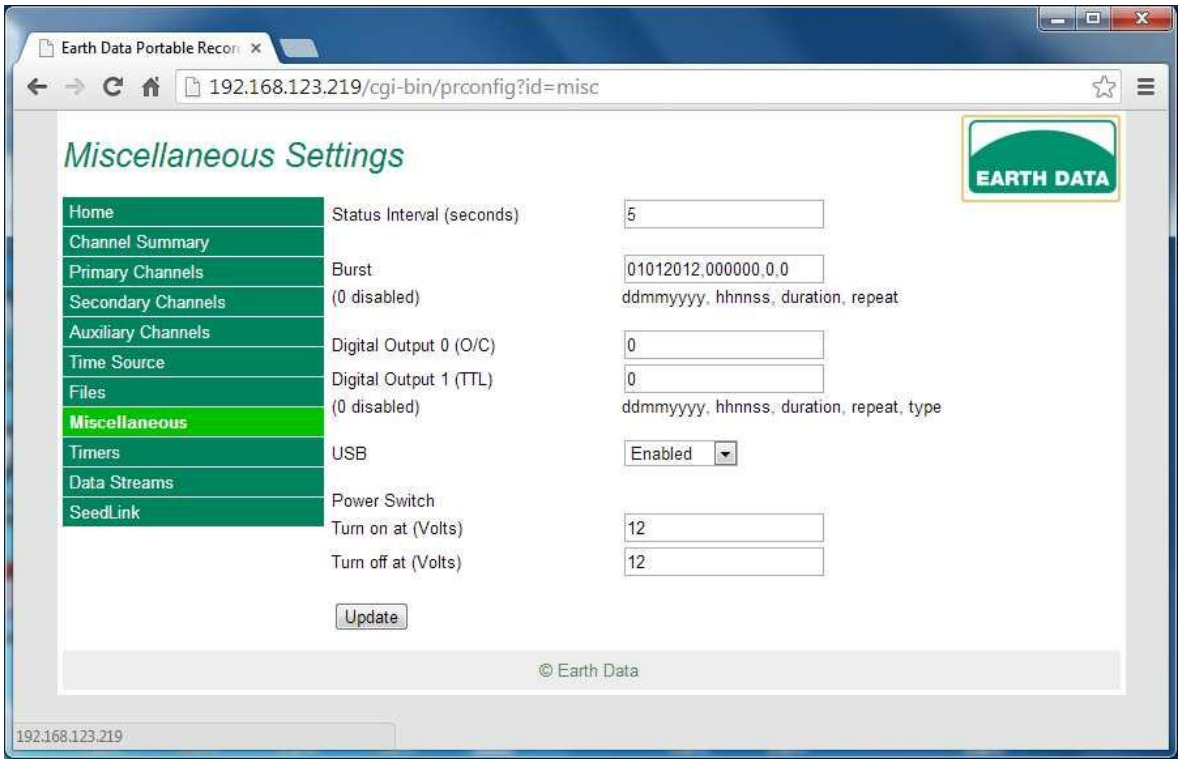
Parameter	Description
Align Files	This entry when ticked aligns the data files to the hour. This is really only to make files visually easier to read. For example where the file size is 10 minutes files will start at 10,20,30,40,50 minutes and on the hour. Specifying files to be an hour long will produce files exactly on the hour.
Delete Old Data	When the disk is deemed to be full this entry directs that either no more data is saved to disk or the oldest data (the oldest day) will be deleted to make space available. Ticking this entry causes the oldest data to be deleted when the disk becomes full. By not ticking this entry you are instructing the recorder to stop recording data when the disk becomes full.
Disk Full Size (MB)	This entry defines when the hard disk is deemed to be full and either no more data will be recorded or old data will be deleted to make space available. The value is specified in Mbytes in the range 1 to 1000 MB. The default when not specified is 50MB, i.e. when 50MB of free space is all that is left the disk is deemed to be full.
Message Log Size (KB)	This entry limits the size of the recorder log file. The value is specified in KB. For example a value of 10000 limits the log size to 10000*1024 bytes, which is just over 10MB. The default size when not specified is 5MB.
Log	This entry enables the recorder log to be saved in its respective day directory. The value can be '**Single**' which keeps a single log file in /data/log/recorder.msg or '**Daily**' which keeps a log file on a day-to-day basis. The default when not specified is to keep a single log file. The log file is named with the same convention as the data files but with hhmmss set to zero. i.e. */data/day_number/yymmdd000000.msg*. The day_logs entry also defines where any trigger log files reside. When day_logs is not specified or 0 any trigger log data is placed in /data/log/recorder.trig. When specified trigger logs are placed in the day directory with the name format as */data/day_number/yymmdd000000.chn.trig* where "chn" is the logical channel name.

Parameter	Description
Storage	**Master Storage** – The primary storage medium – can be **Earth Data USB HDD or SSD**, **Earth Data USB CF** (Compact Flash), **Earth Data USB 4-in-1** (e.g. SD), **Internal CF**,(Internal Compact Flash **Other USB** (e.g. USB Pen Drive) **None**. **Secondary Storage** devices can be set for mirroring the data but is currently not implemented.
Max Flush Interval (mins)	The maximum number of minutes that the unit will hold data (in memory) before committing it to the selected storage media.

Recommended Mini Seed File Size Limits*

Sample rate samples/second	Maximum Mini-SEED file size in minutes
3000	10
1000	30
750	40
600	50
500	60
375	80
300	100
250	120
150	200
125	240
120	250
100	300
75	400
50	600
25	1200
20	1440
10	1440
5	1440
4	1440
2	1440
1	1440

2.6.2.7 Miscellaneous

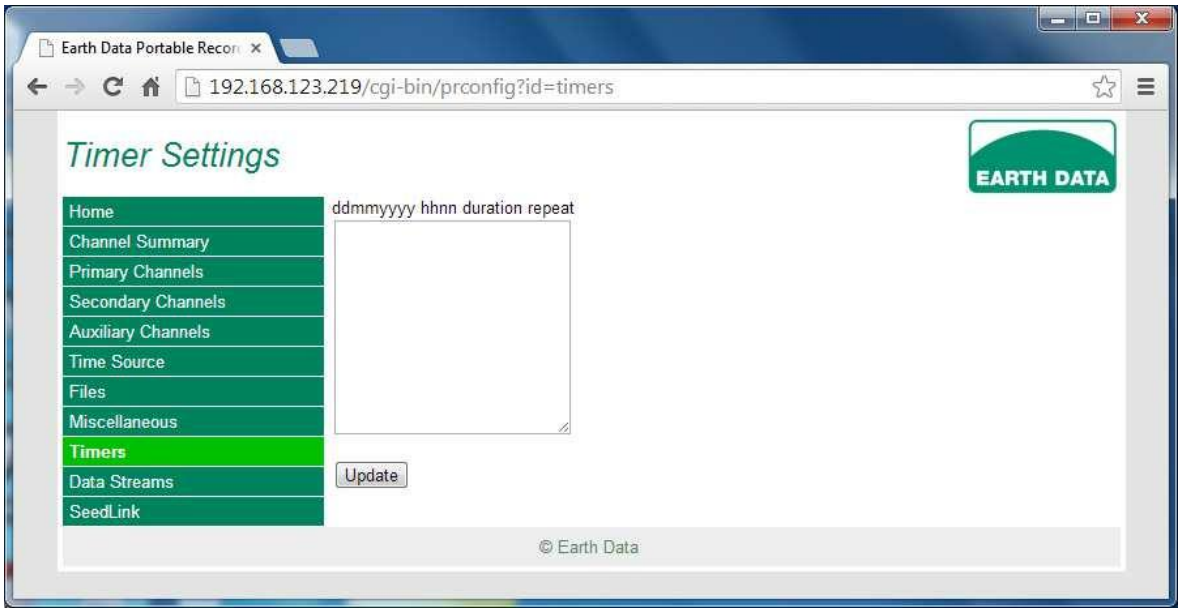


Parameter	Description
Status Interval	This entry defines how often the live recorder status is sent. The value is specified in seconds in the range 1 to 60. The default when not specified is 5 seconds.
Burst*	The arguments for this entry are as specified below: **ddmmyyyy, hhnnss, duration, repeat** This entry defines that the primary data channels are to operate in burst mode. Recording on the primary channels will commence at the time and date specified for the **duration** in minutes. The argument **repeat** allows the burst sequence to be repeated every **repeat** minutes. The date and time specified must be in this century. A value of zero disables burst mode.

Parameter	Description
Digital Output 0 (O/C) Digital Output 1 (TTL)	The arguments for this entry are as specified below: **ddmmyyyy, hhnnss, duration, repeat, type** This entry defines the state of the digital output line d0. This output is an open collector driven output. The output is asserted (active) or not asserted (at rest). The output is asserted at the time and date specified for the **duration** in seconds. The **type** specifies the state when asserted and at rest, **type** has two states as shown below: **type=0** Rest state is high impedance, single pulse of low impedance for **duration** **type=1** Rest state is low impedance, single pulse of high impedance for **duration** **repeat** allows the calibrate sequence to be repeated every **repeat** minutes. The date and time specified must be in this century. A value of zero disables Digital Output.
USB	**Enabled** – USB interface will be cycled as and when required. **Always On** – USB interface is always powered. **Disabled** – USB interface is never powered.
Power Switch Turn on at (Volts) Turn off at (Volts)	These fields program the input lower voltage level (Turn on at) that allows the unit to power on and the voltage that the unit will power off at (Turn off at).

*Burst mode recording allows high sample rate data to be collected for short periods of time whilst recording continuously at a lower rate. This is a special mode of dual rate recording. The secondary data channels (channels 6 to 11) record data continuously while the primary channels (channels 0 to 5) record data at a higher rate but only for periods specified by this entry. Each channel is configured as normal for dual sample rate recording together with this additional setting specifying that the primary channels operate in burst mode recording. Using the "burst" entry recording on the primary channels is switched on and off either as a one-off operation or repeated at intervals.

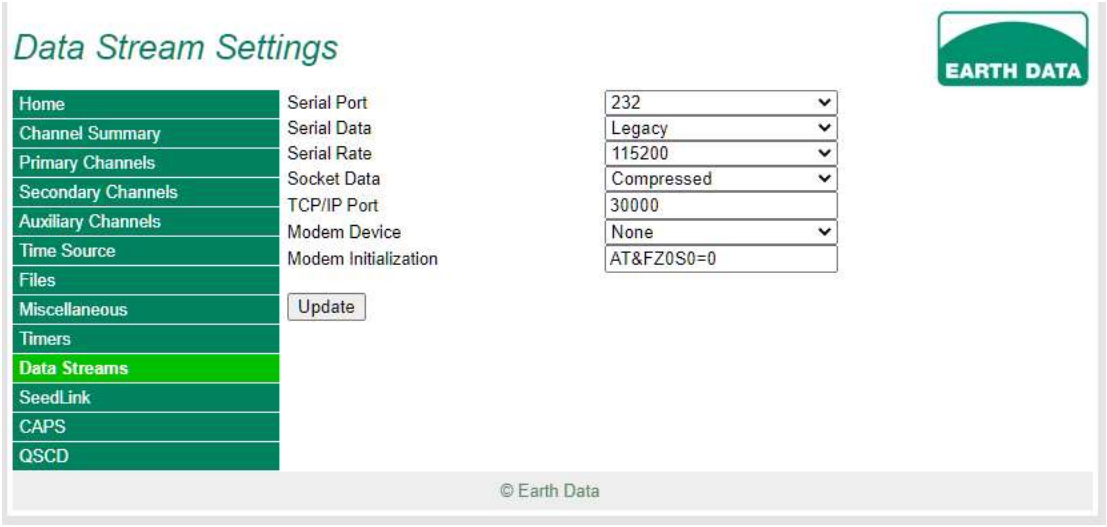
2.6.2.8 Timers



Parameter	Description
Timer Settings	The arguments for this entry are as specified below: **ddmmyyyy hhnn duration repeat** This entry defines a timer window. The timer settings are a list of one or more lines describing the recording pattern for the unit. In the absence of any entries the recorder records continuously. The value **repeat** is optional. Up to 200 entries can be made at any one time, all specifying a different start date and duration. A single entry can be made with a **repeat** field that will cause the recorder to record data at a specific time each day or week. The fields **duration** and **repeat** are specified in minutes. For example the entry '*01082002 1400 60 480'* will cause 60 minutes to be recorded every 480 minutes (8 hours). That is to say three times per day at 14:00, 22:00 and 06:00. If the date and time specified by *ddmmyyyy hhnn* is in the future the recorder will stay off until that future date and time. As that date and time becomes history the **repeat** value is used to calculate either the end of a current session or the beginning of the next session. Note.. Currently timer windows are not implemented.

2.6.2.9 Data Streams

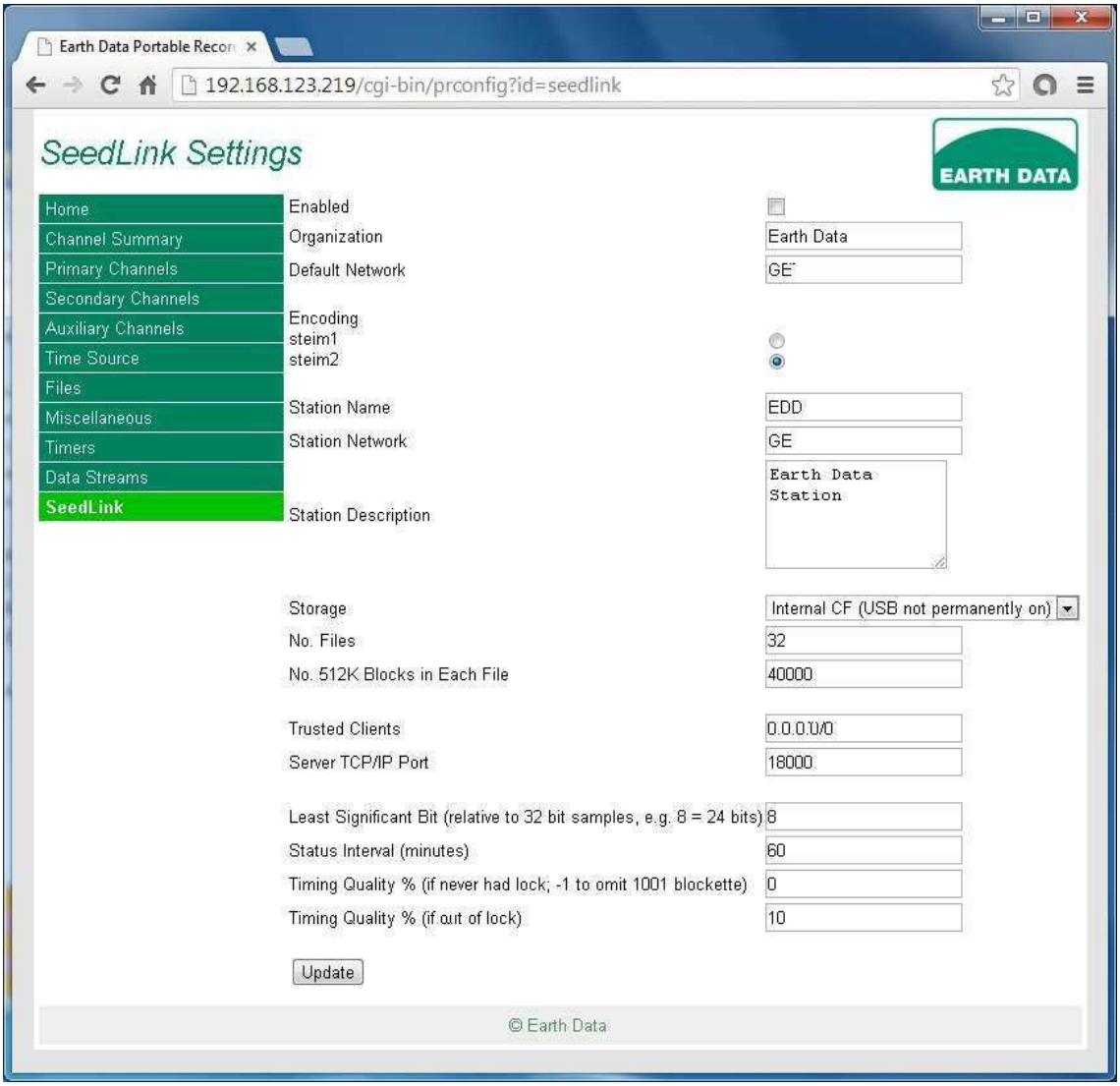
Select 'Data Streams' to configure the format for the communication streams. The display will be similar to that shown below.



Parameter	Description
Serial Port	Select between 232 and 485 MODMON serial output
Serial Data	This entry defines the format for the serial data communications: **Legacy:** standard MODMON format. **Compressed:** compressed MODMON format. **None**: Serial data is disabled.
Serial Rate	Serial data baud rate (2400...230400).
Socket Data	This entry defines the format for TCP/IP serial data streaming: **Legacy:** standard MODMON format. **Compressed:** compressed format. **None: TCP/IP data is** disabled.
TCP/IP Port	The Ethernet TCP/IP communications port number

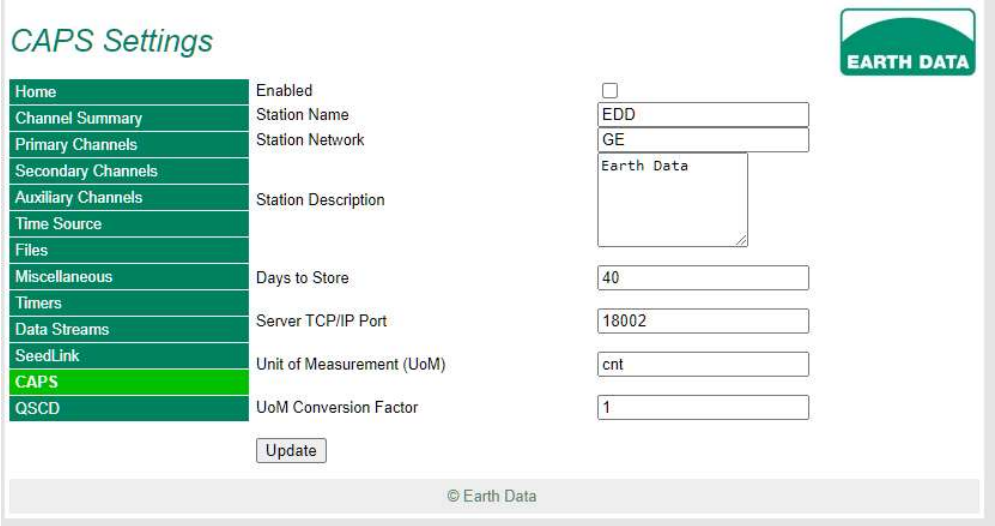
Parameter	Description
Modem Device	This entry is set if a modem is connected to one of the available communication ports, valid settings include: **None:** No modem is connected. **ttyS1:** Modem connected to RS232 serial port. **ttyACM0:** Connected USB Modem supports standard Communication Device Class Abstract Control Model (CDC ACM) Driver. **ttyUSB0:** Modem is connected via supported USB to serial adapter (e.g. using FTDI or Prolific chipsets).
Modem Initialization	This 'AT' command set is used to initialize the modem by the inbuilt 'mgetty'. The default setting for this is 'AT&FZ0S0=2' - i.e. reset, restore defaults and set to answer on 2 rings.

2.6.2.10 SEEDLink



Parameter	Description
Enabled	Select or de-select in order to enable, disable the SEEDLink server respectively.
Organization	Organisation ID, shown with HELLO request or slinktool -I. (Arbitrary string.)
Default Network	Default network code; used when a network code is omitted by a client in STATION request. 1 or 2 characters long, uppercase.

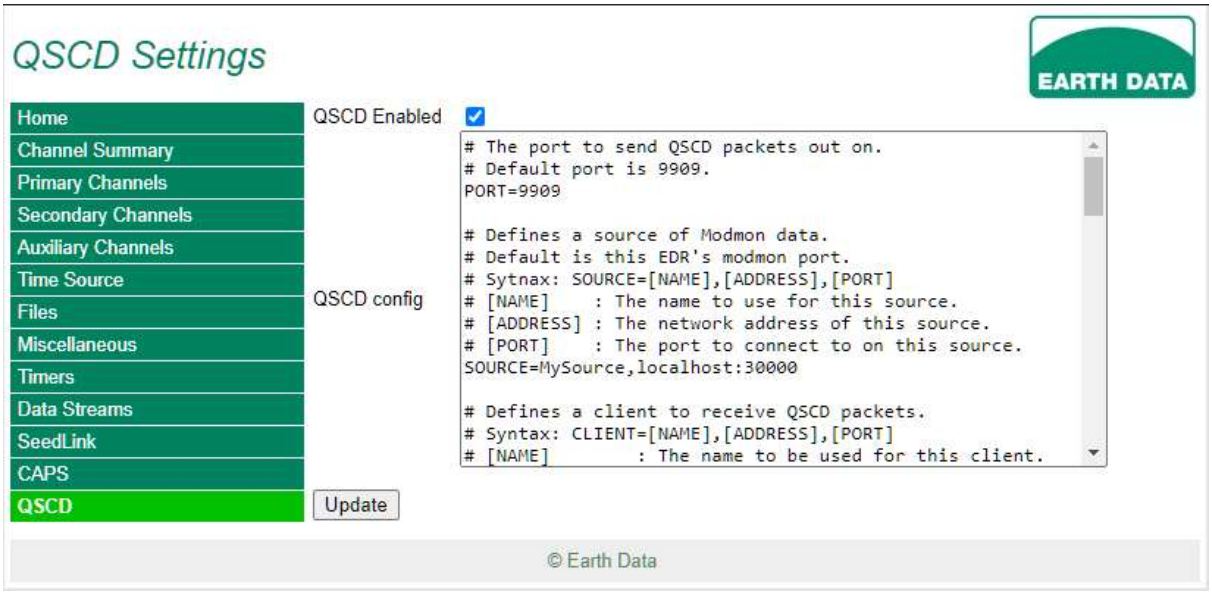
2.6.2.11 CAPS



Parameters	Description
Server TCP/IP Port	Server TCP/IP communications port number
Unit of Measurement (UoM)	Unit of measurement for counts in counts data
Unit Conversion Factor	This is the value multiply with unit of measurements

2.6.2.12 QSCD

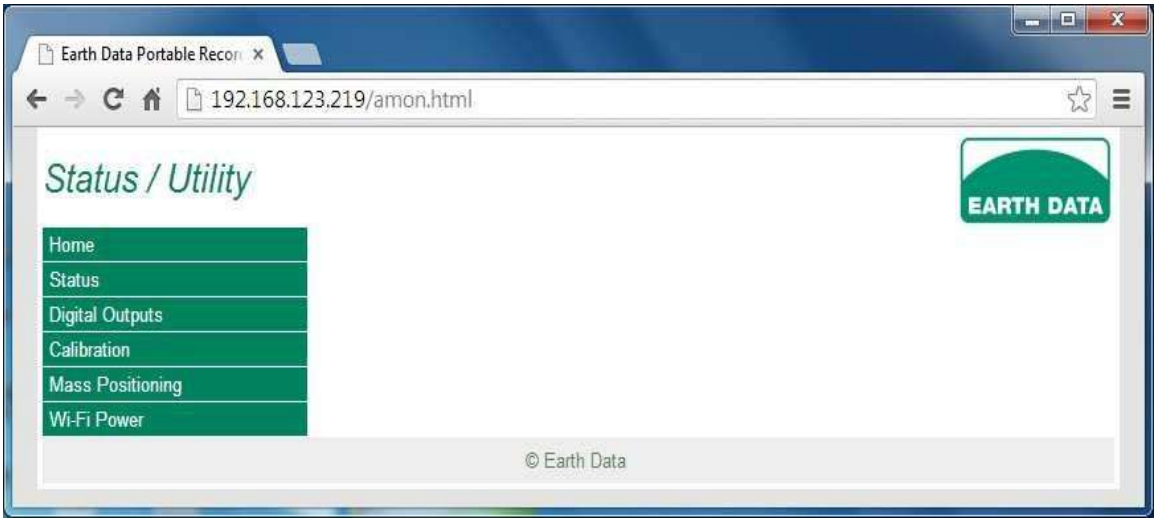
Select ‘QSCD Enabled’ to enable the QSCD protocol and display the configuration box.



Parameter	Description
Encoding	Encoding when converting raw streams to Mini-SEED. Options are: "steim1" or "steim2". Note.. Currently only steim1 is supported.
Station Name	Station code, 3 characters long
Station Description	Descriptive text for station
No. Files	Number of temporary files to keep on disk (ring buffer).
No. 512K Blocks in Each File	Number of 512K blocks per file.
Trusted Clients	Restrict any SEEDLink output to selected trusted clients (IP addresses).
Server TCP/IP port	SEEDLink TCP/IP port, defaults to 18000.
Least Significant Bit	Least significant bit relative to 32-bit samples. Normally set to 8 for 24-bit samples, but can be set, for example, to 7 to get 25-bit samples.
Status Interval (minutes)	Interval in minutes when "state of health" information is logged, "0" means disabled.
Timing Quality (No lock ever)	Default timing quality as a percentage. This value will be used when no timing quality information is available. Can be -1 to omit the blockette 1001 altogether.
Timing Quality (out of lock)	Timing quality to use when GPS out of lock.
Storage	Storage medium for the SEEDLink buffer, can be: Internal CF if USB is cycled, otherwise if USB is always on it can be a USB storage device.

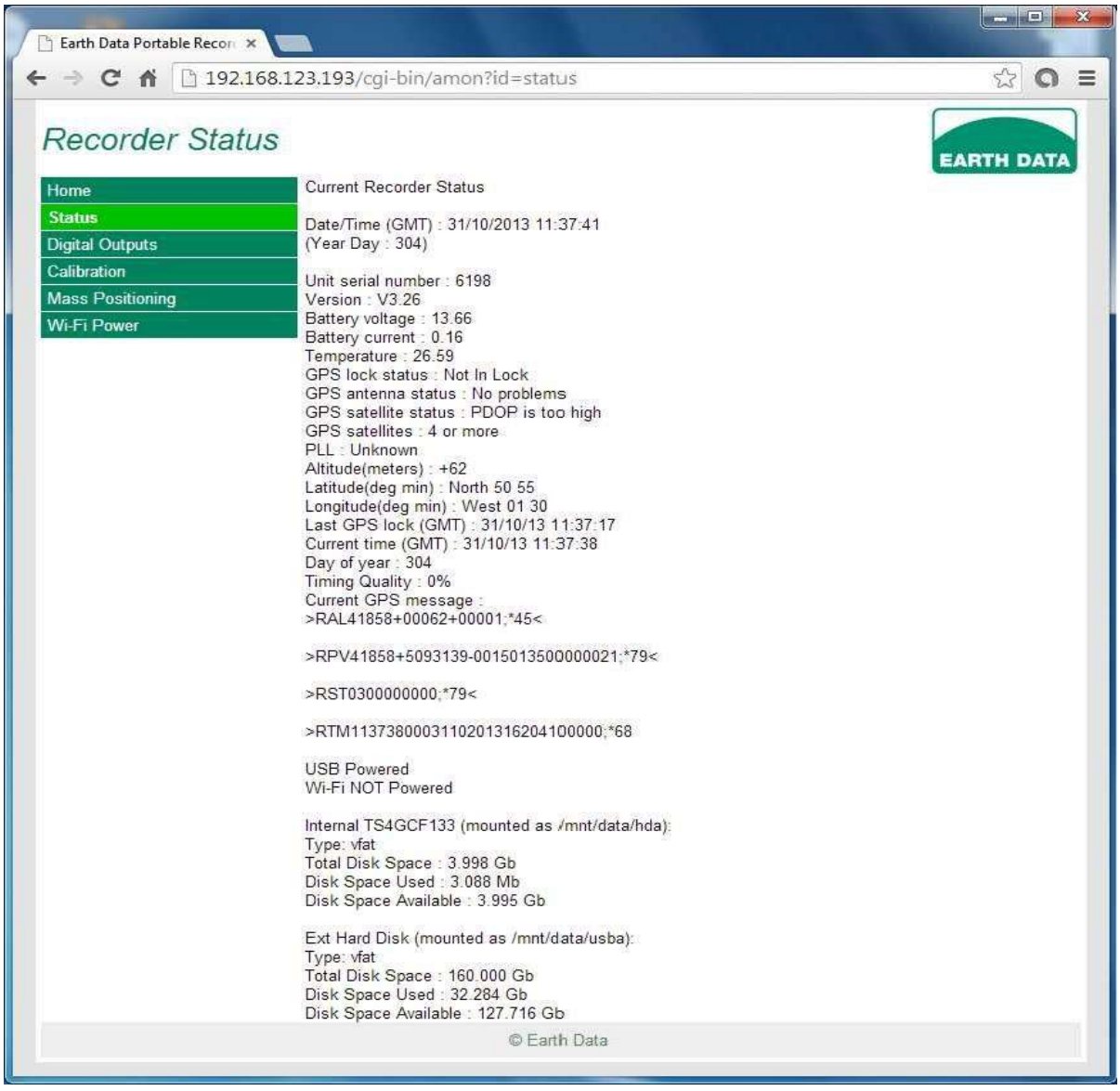
2.6.3 Status & Utility

Select 'Status / Utility' from the main page to view the unit's status and operate some of the utilities such as sensor calibration or mass positioning.



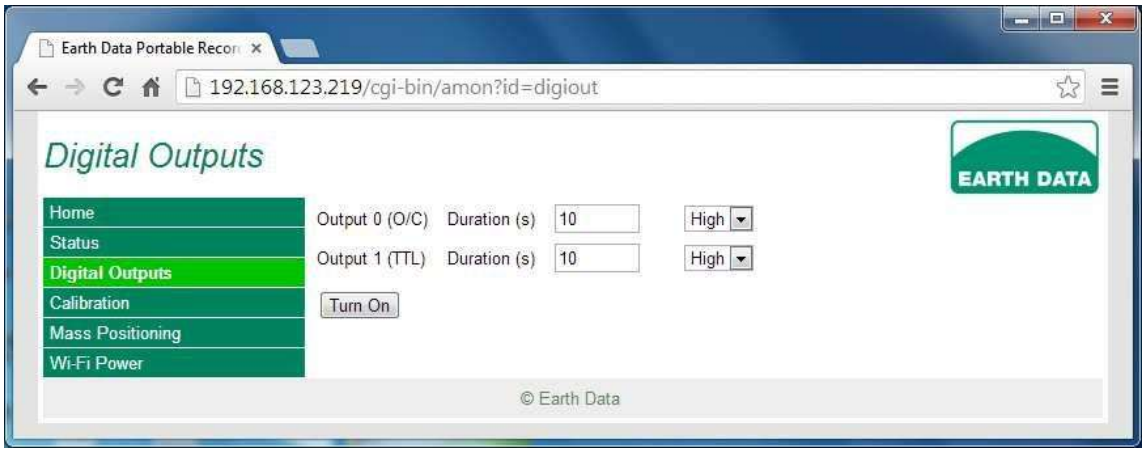
2.6.3.1 Recorder Status

Select Status to view the main unit status and health indicators.



2.6.3.2 Digital Outputs

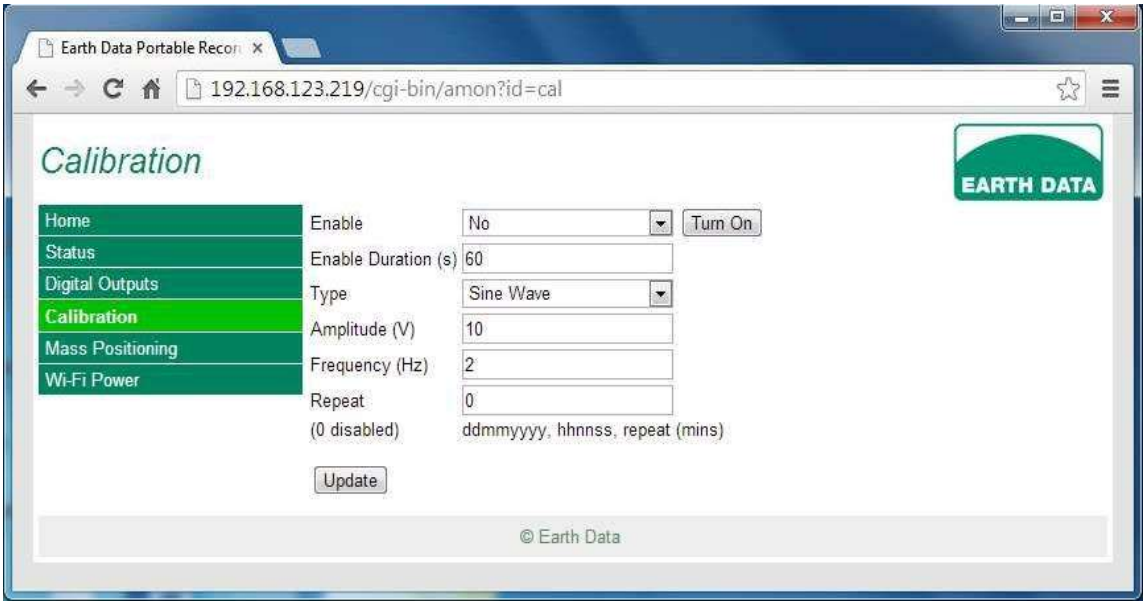
The unit is fitted with two digital outputs. Digital Output 0 is an Open Collector type (O/C) whereas Output 1 is a TTL type.



Click the “Turn on” button to pulse (High or Low) one or both of the digital outputs for duration seconds.

2.6.3.3 Calibration

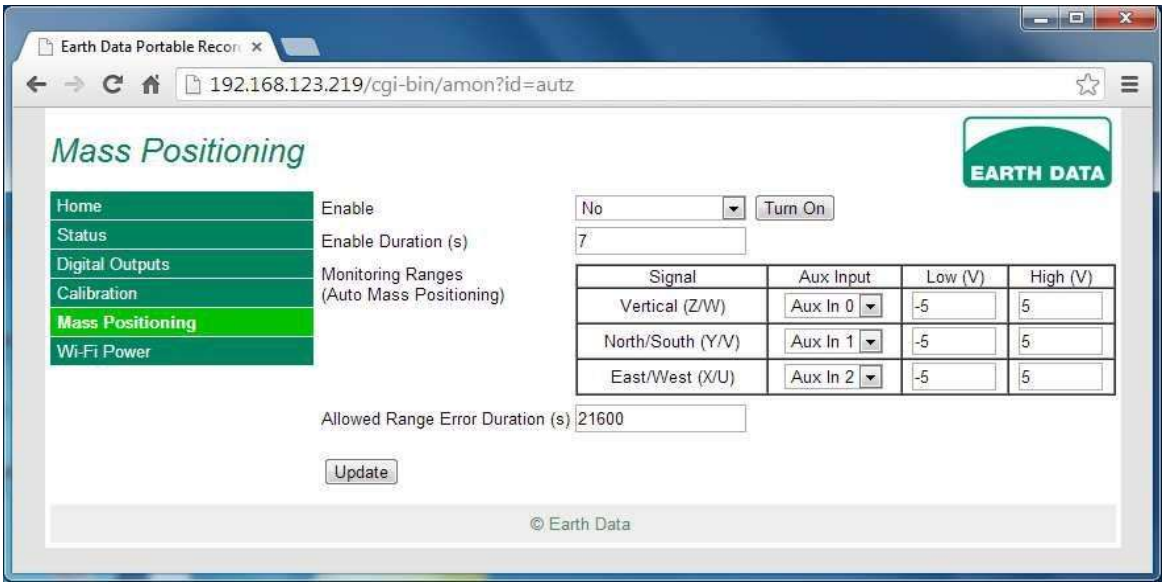
In order to calibrate the connected sensor, or configure periodic calibrations of the connected sensor you will need to enter parameters using the calibration interface. Sensor specific cables are required to carry out the calibration, some of which are available from Earth Data.



Parameter	Description
Enable	**No**: Unit calibration disabled. **Digital 0 (O/C) Hi**:- Digital 0 (O/C) driven High **Digital 0 (O/C) Lo**:- Digital 0 (O/C) driven Low **Digital 1 (TTL) Hi**:- Digital 1 (TTL) driven High **Digital 1 (TTL) Lo**:- Digital 1 (TTL) driven High
Enable Duration (s)	Number of seconds that the calibration is enabled for.
Type	**Sine Wave**: sinusoidal calibration signal. **Square Wave (Pulse)**: pulsed calibration signal. **White Noise**: white noise calibration signal.
Amplitude (V)	Calibration signal voltage amplitude.
Frequency (Hz)	Calibration signal frequency – only applicable for sine or square wave inputs.
Repeat	If enabled the calibration is repeated from ddmmyyyy, hhnnss every **repeat** minutes. For example 01012012,140000,1440 repeats the calibration at 14:00:00 every day (1,440 minutes) from 01/01/2012.

2.6.3.4 Mass Positioning

In order to mass position the connected sensor, or automatically monitor the mass position of the connected sensor you will need to enter parameters using the mass positioning interface. Sensor specific cables are required to carry out the mass positioning, some of which are available from Earth Data. The monitoring is achieved using the auxiliary inputs.

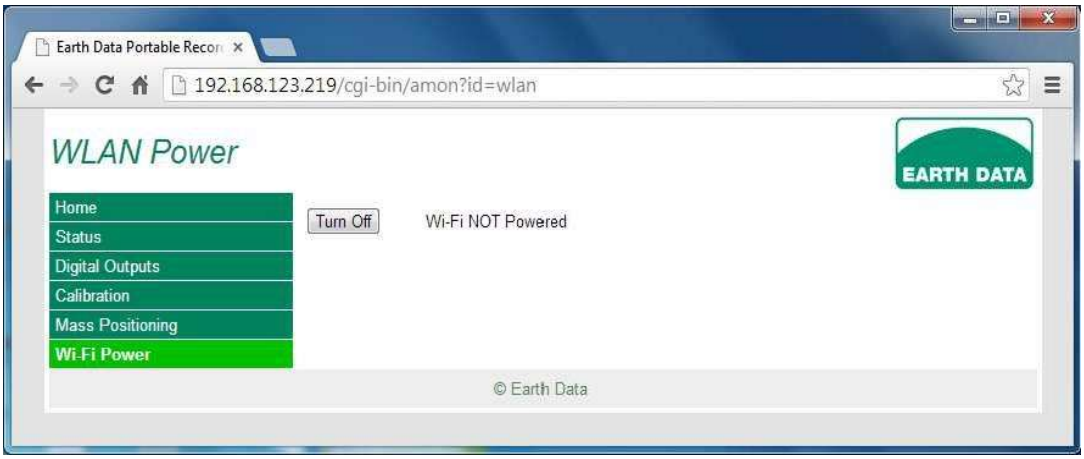


Parameter	Description
Enable	**No**: Unit mass positioning disabled. **Digital 0 (O/C) Hi**: Digital 0 (O/C) driven High **Digital 0 (O/C) Lo**: Digital 0 (O/C) driven Low **Digital 1 (TTL) Hi**: Digital 1 (TTL) driven High **Digital 1 (TTL) Lo**: Digital 1 (TTL) driven High
Enable Duration (s)	Number of seconds that the mass positioning enable signal is set for.
Aux Input	Auxiliary channel monitoring the mass positions for the Vertical (Z/W), North/South (Y/V) and East/West (X/U) axes.
Low (V)	Minimum voltage (for respective axes) at which Auto Mass Positioning will be triggered. In the example above any axes measuring less than -5 Volts for 6 hours (21,600 s) will trigger an auto mass position.

Parameter	Description
High (V)	Maximum voltage (for respective axes) at which Auto Mass Positioning will be triggered. In the example above any axes measuring greater than +5 Volts for 6 hours (21,600 s) will trigger an auto mass position.
Allowed Range Error Duration (s)	Period in seconds for which the axes monitoring voltage can be out of range before an auto mass position is initiated.

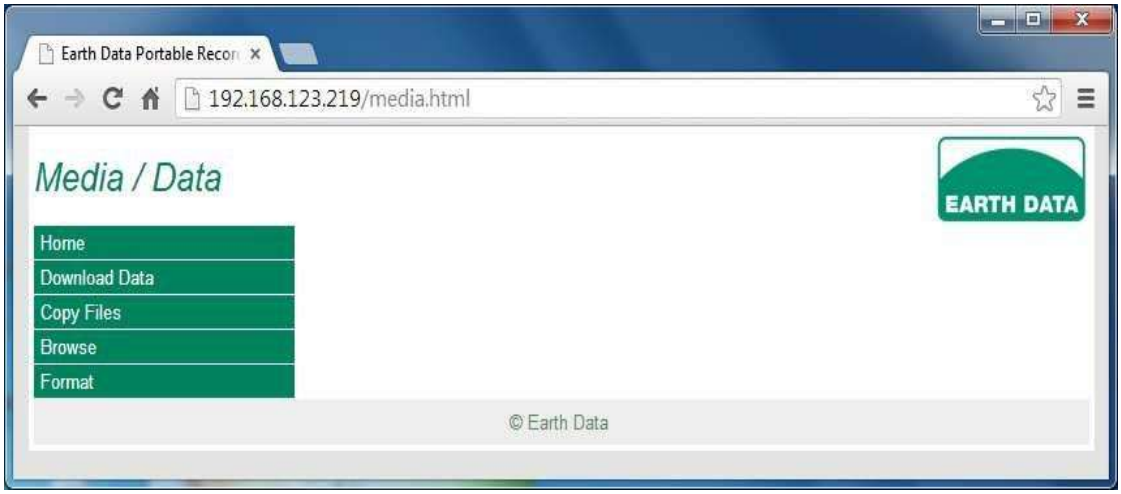
2.6.3.5 WiFi Power

This allows the WiFi to be forced to be powered off.



2.6.4 Media / Data

Select Media / Data from the main page to access any media connected via the USB connection(s) such as the removable disks, USB Pen drives, USB Flash Card Readers, or the internal Compact Flash.



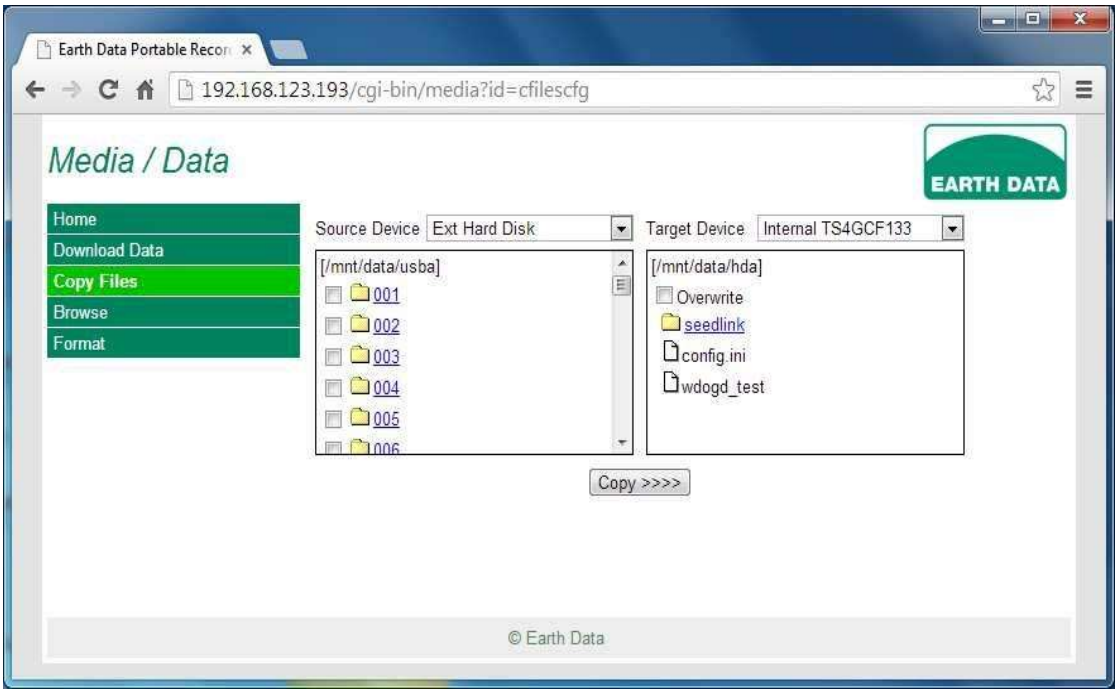
2.6.4.1 Download Data

This allows stored Mini-SEED or ASCII data files to be downloaded to a host.



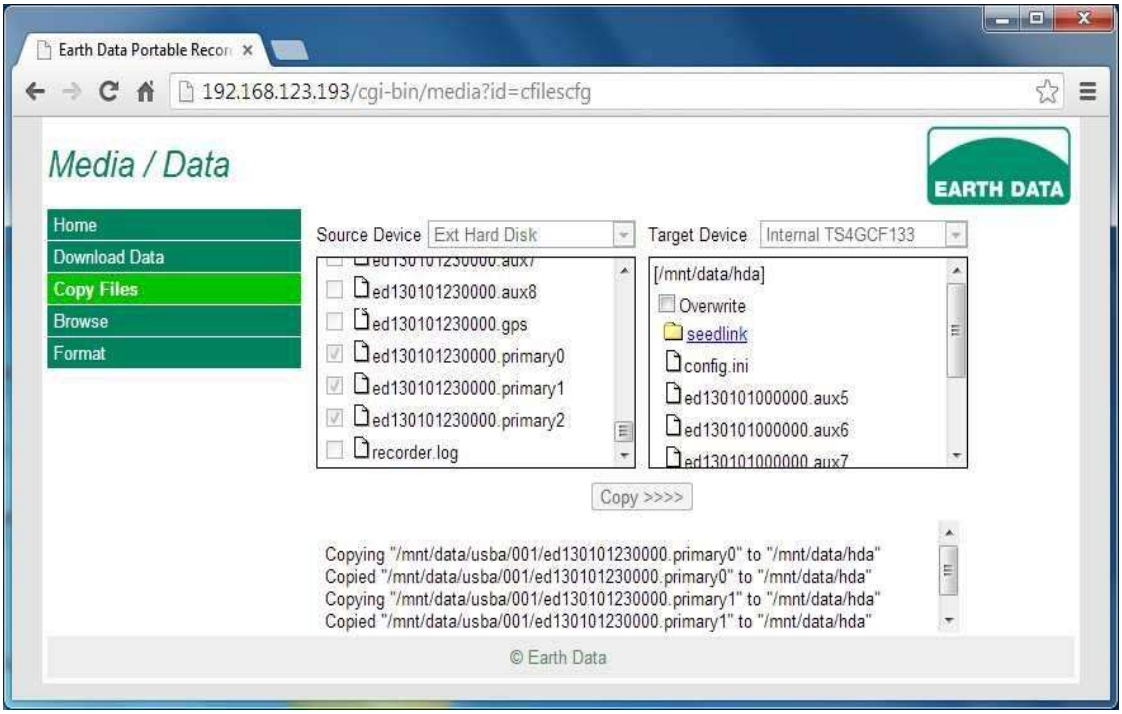
2.6.4.2 Copy Files

Select ‘Copy Files’ to copy recorded data on to the removable media.



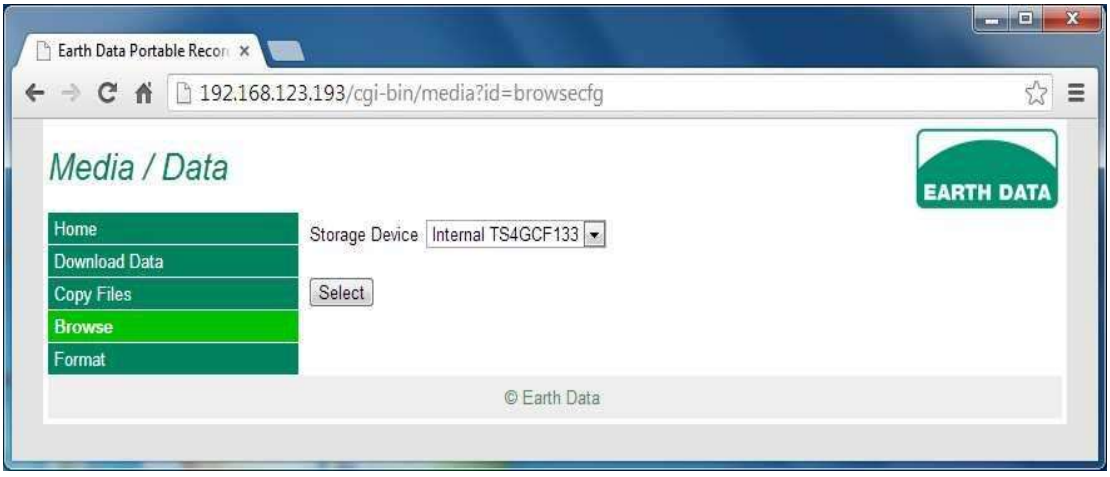
Parameter	Description
Target Device	This lists the available Removable Devices currently connected. Simply select the device that is to be copied to.
Select Start Day Select End Day	All available recorded day(s) are displayed. Data with a date greater than or equal to the Start Day and less than or equal to the End day will be copied to the target device.
Destination Folder	By default each day will get written into its relevant year day folder (e.g. ‘170’) off the root of the target device. However, by entering a ‘Destination Folder’ each day will get written into its relevant year day folder (e.g. ‘170’) off the Destination folder of the target device. For example, entering ‘recorder1’ for the destination folder would result in day ‘170’ data getting written to the directory ‘/recorder1/170’. This is useful if you are gathering data to the same piece of removable media from more than 1 recorder.

The browser window informs you as the copy starts and finally completes.



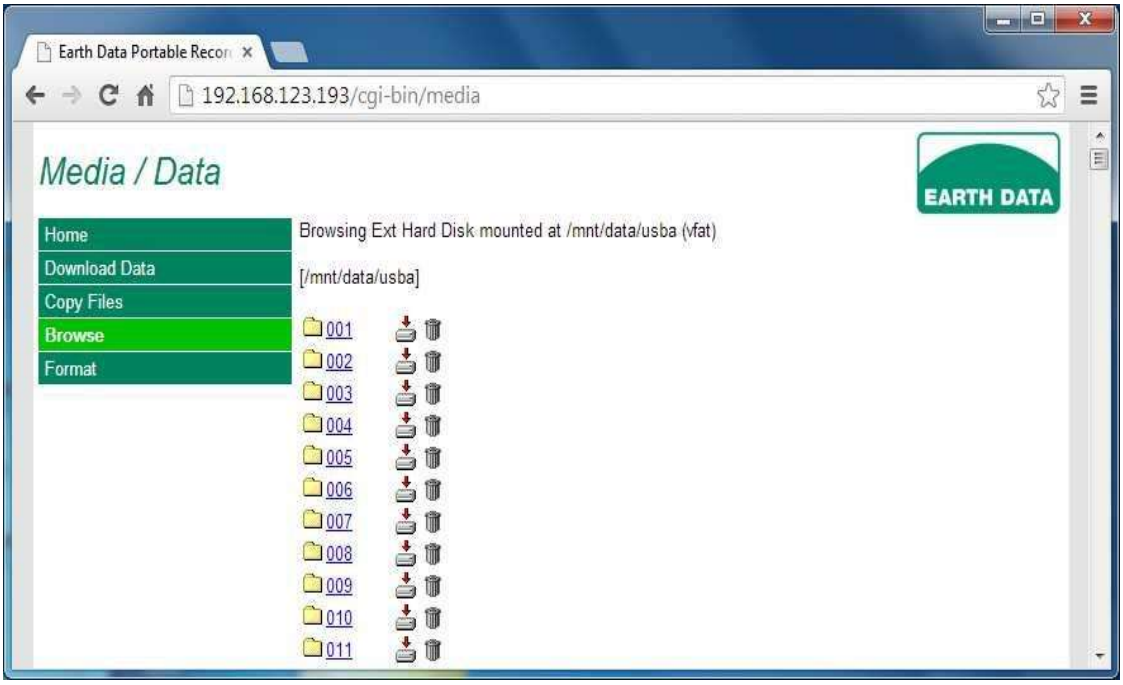
2.6.4.3 Browse

Select 'Browse' to browse the contents of a storage device and optionally download or delete files and folders from it.

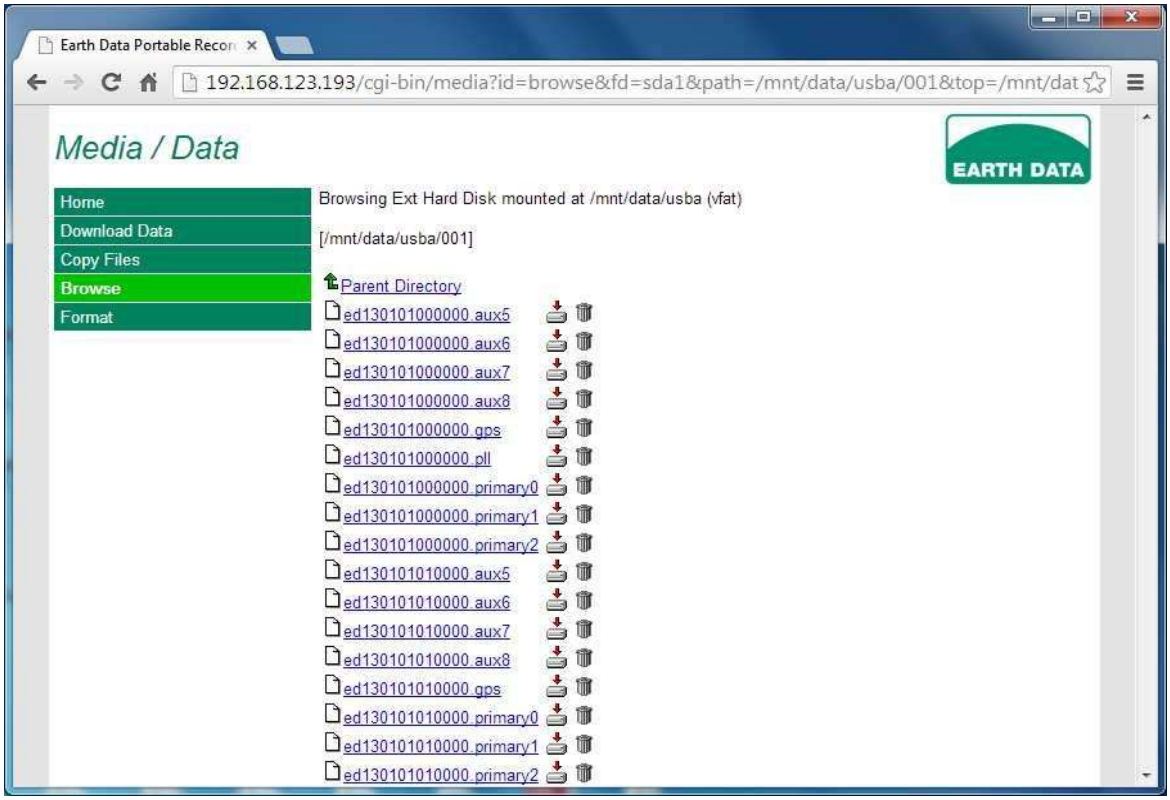


Select the Storage Device that you want to browse.

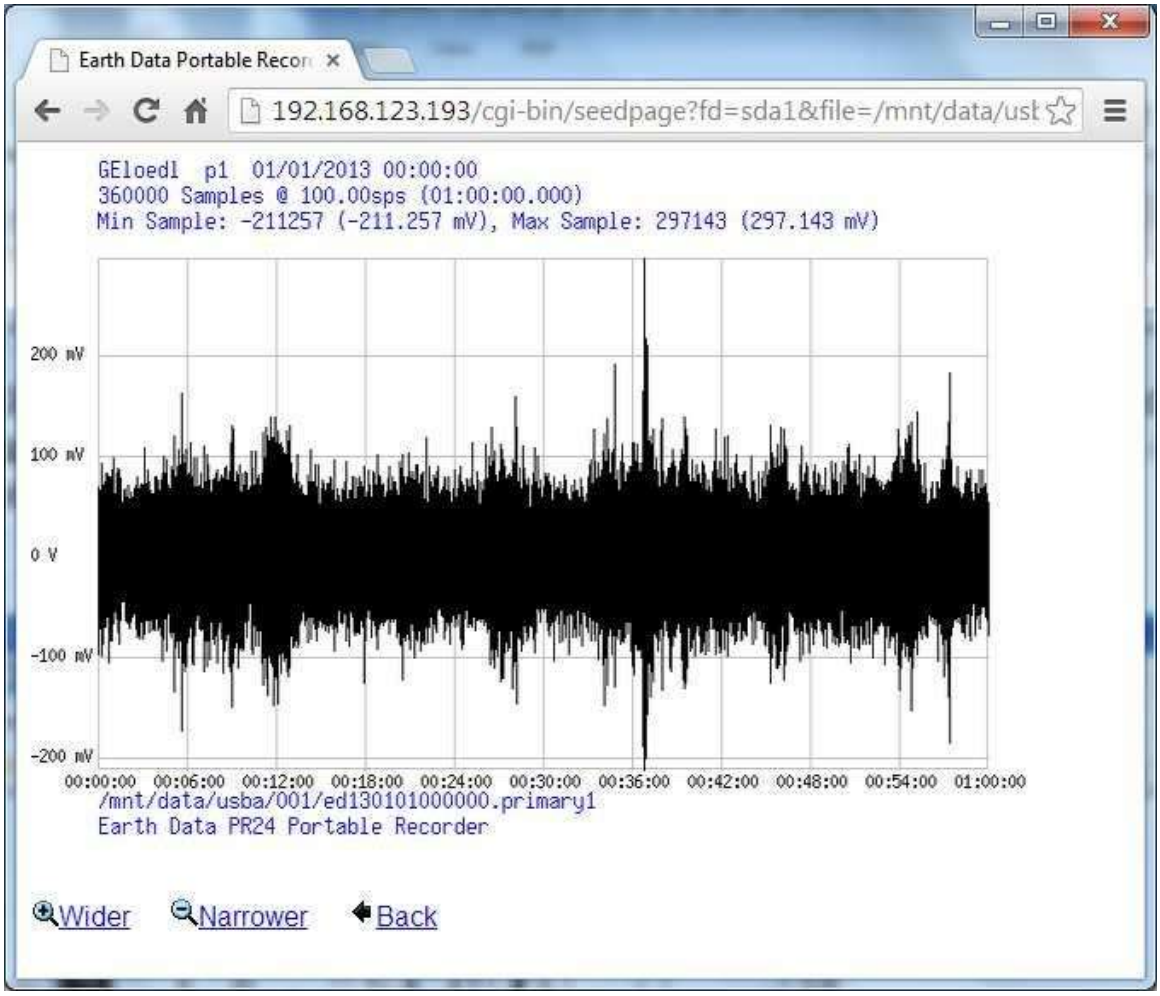
Each day's worth of data is stored in a day directory, select a day (relevant to the day of the year) entering a day directory displays all the relevant data recorded for that given day.



Folders are displayed as links that you can drill down into, whereas files are simply displayed as labels. Whole folders and individual files can be downloaded or deleted by clicking on the relative ‘download’ and ‘trash’ icons respectively.

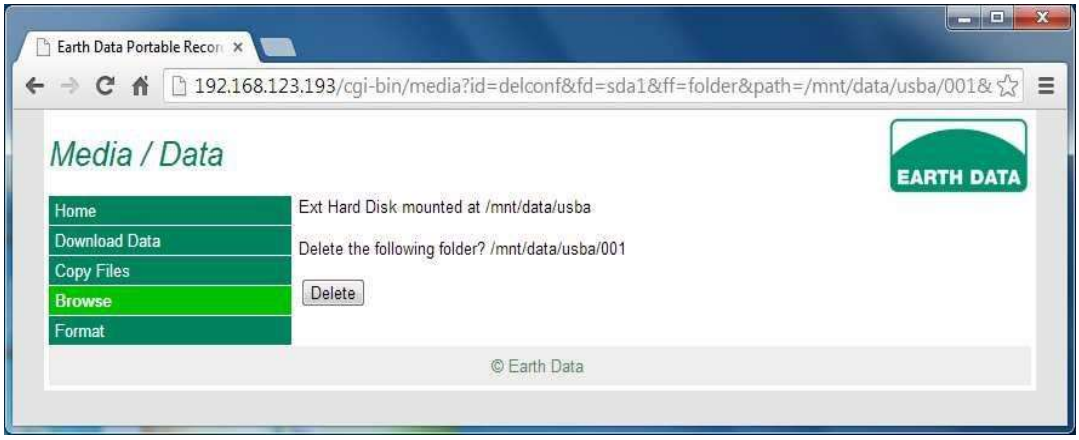


Click on a data file (e.g., *.pri1 in this case represents channel 1 data) to view the data or click the download icon to save it. Viewing a file displays it within the browser window as shown.



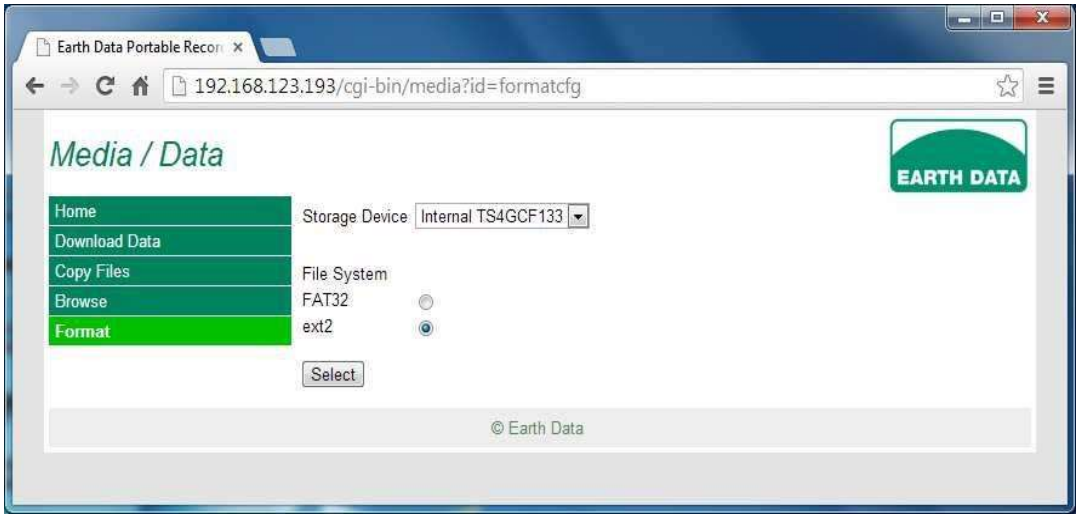
Note that, you can increase / decrease the viewing resolution by selecting 'Wider' and 'Narrower' respectively.

Note that when you select to delete a file / folder you are prompted for confirmation.



2.6.4.4 Format

In order to format a storage device select 'Format'. If you ever intend to use the storage on a Windows PC choose FAT32. If you live entirely in the Linux world choose ext2. If formatting the internal compact flash for use with SEEDLink then it is recommended that the ext2 file system is employed.

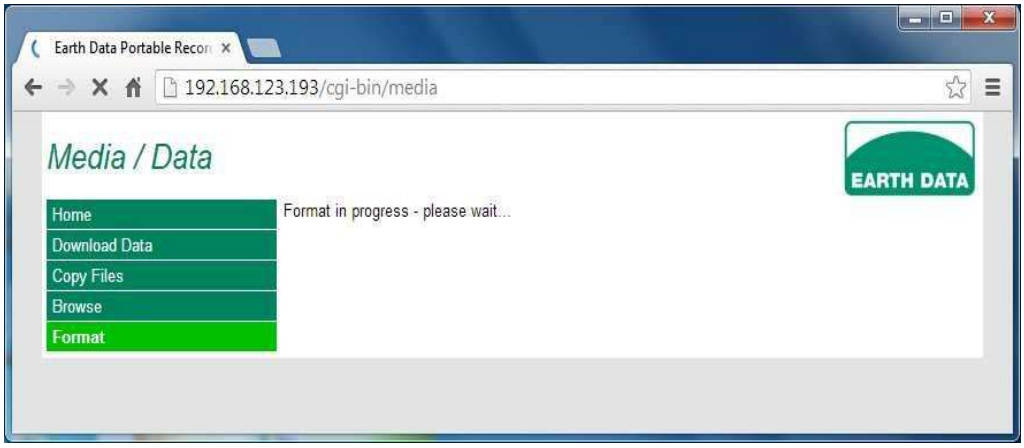


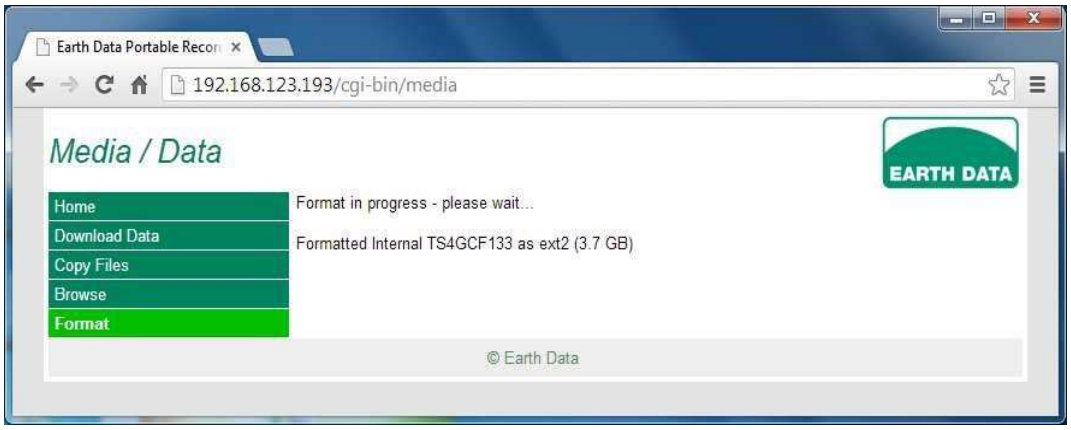
Parameter	Description
Storage Device	This lists the available Storage Devices currently connected. Simply select the device that is to be formatted.
File System	Choose either 'FAT32' or 'ext2'. Note that if you wish to be able to read the device on both a Linux and Windows system then FAT32 is probably the best option. The most secure option is ext2. Use this if you are likely to read a device on a Linux system.

You will be prompted for confirmation before formatting commences.



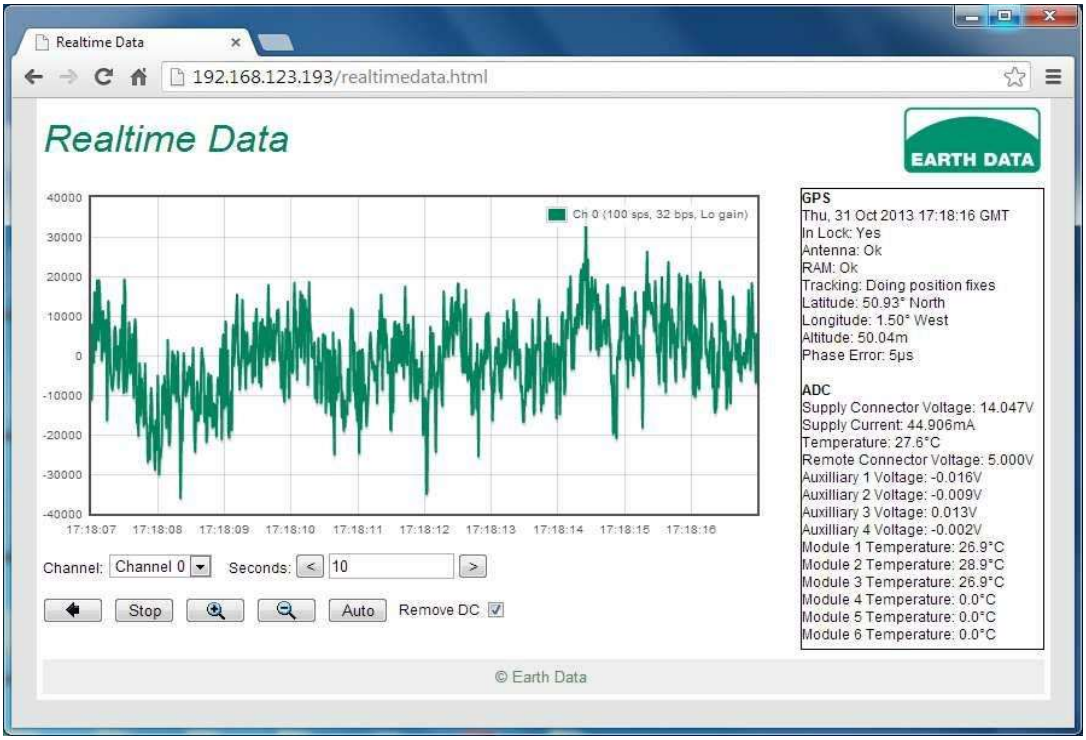
The browser window informs you as the format starts and finally completes.





2.6.5 Real-time Data

Select Realtime Data from the main page to view realtime data, this is a useful diagnostic during installation to ensure everything is running ok.



Parameter	Description
Channel	Channel to be displayed (0…5).
Display Second(s)	The number of data seconds to display in scrolling window.
Scale	Click Auto to automatically scale or magnify to manually scale.
Remove DC	Click to remove any DC offset.

2.6.6 Firmware Update

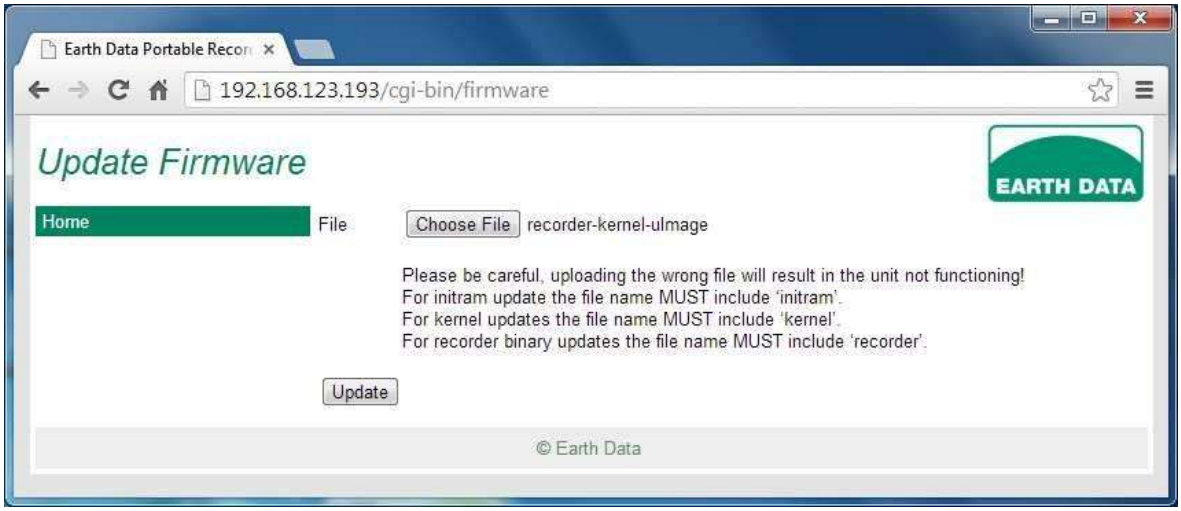
Select Firmware Update from the main page in order update the recorder's kernel, initram or recorder software via the web interface.



Choose the relevant file, using the indicated button. Note that in order to update the 'initram' the 'initram' file name will include 'initram', similarly kernel update will include 'kernel' and recorder binary update will include 'recorder' in the file names respectively.

The chosen a file's name will be displayed on the web page.





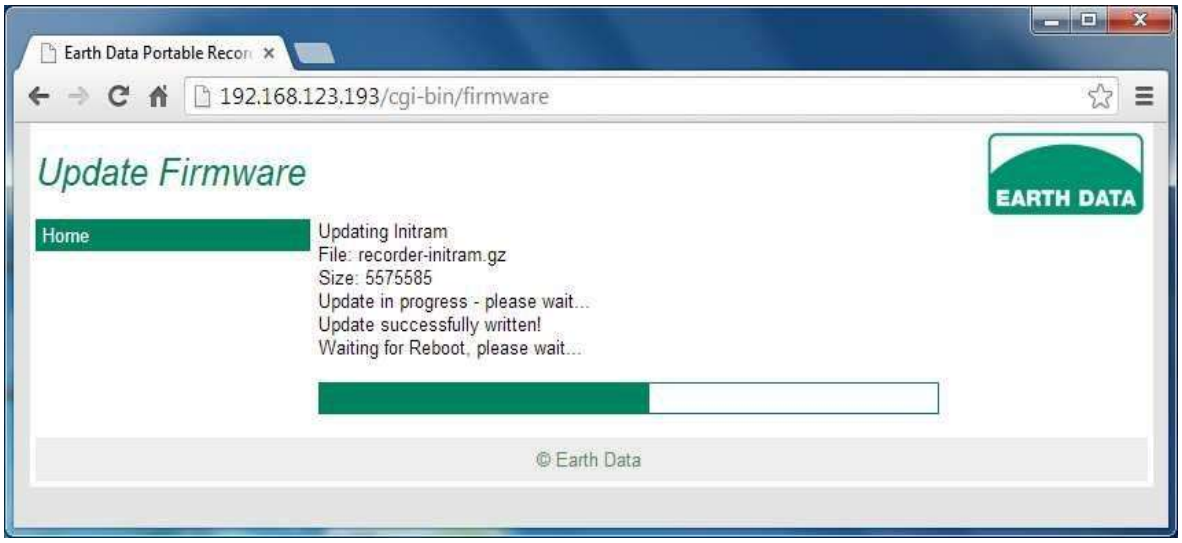


Press the Update button in order to proceed and apply the relevant update. A confirmation box warning about whether to proceed will be indicated.

Having confirmed, the web page uploads the selected files and indicates size of the file being applied the progress of the update as it is committed to the unit's onboard FLASH.



Once completed, a confirmation box asking whether to re-boot now will be displayed and if re-booted the web page will display the progress of the re-boot.



2.7 MODMON

A functional check of the digitizer can be carried out by using the the MODMON32 utility normally supplied with the unit. This can be over the RS232/485 port or Ethernet.

Download MODMON32 from our website, run, and observe a green indicator light in top left corner of the screen. There will be a 10 or 20 second delay after switching on the digitizer while it runs through its boot sequence. This light indicates that valid one-second packets are being received from the digitizer. When the light starts to flash the main screen will begin

writing the time series data for each of the first three channels. To view data for all six channels enable the enhanced header from the Control page, this may require a higher Baud rate.

If required a signal source can be connected and viewed in near real-time on the MODMON32 main screen which is designed to mimic an oscilloscope. Use the red + and - buttons to increase or decrease the sensitivity of the display and the "DISPLAY SECONDS" box to alter the timebase. Alternatively use the U and D keys on the keyboard to change the sensitivity and Z and X to change the timebase. Use the "remove DC" to a.c. couple the display. An overlay button allows the display to act like a storage scope. When using high sensitivity it can be useful to stop and start MODMON using "START" and "STOP" buttons on the toolbar, since this initialises the a.c coupling.

Selecting HEADER will allow the current operating parameters to be inspected.
The header includes the digitizer firmware issue and unit serial number.

Position the GPS antenna such that it has a reasonable view of the sky and confirm correct GPS operation by enabling the enhanced header from the CONTROL page, which may require a higher Baud rate, and selecting the GPS page on the MODMON32 utility. As soon as GPS lock is achieved, the messages will be included in the header and written to this page. If the GPS subsequently loses lock then the last valid message will continue to be sent. The age of the last valid message is included in the header so that valid time can still be deduced.

Getting initial lock may take several minutes, up to ½ an hour if the unit has been moved by many kilometres or has not been used for several weeks. Subsequent locks should only take a few minutes because the positional information for the GPS is held in battery-backed ram.

It is strongly recommended that the digitizer be checked in this way on receipt of the unit.

2.7.1 Setting the Input Configuration

The digitizer is despatched ready for operation. All 3 channel units will be configured as 3 channels of 100 samples per second and 0dB gain for each channel, 115,200 Baud. All 6 channel units will be configured as 6 channels of 100 samples per second and 0dB gain for each channel, 115,200 Baud. This will be the case unless specifically requested. The word size is set to 24 bits

Unless the user has written his own control software, the unit can be configured using a Web Browser over the Ethernet.

Always remember that the CONTROL page only shows the settings that will be sent to the digitizer. It does not show the current operating parameters of the digitizer. For this it is necessary to press the HEADER button. Any changes made using the CONTROL page should be checked by looking at the header sent back from the digitizer.

Do not change more than one item at a time. When a parameter change is sent the flashing green light stops for a while until the new parameter has been accepted. It is best to wait for the flashing light to re-appear before changing other parameters.

On the serial RS232 port even at 115kbaud, the digitizer cannot support 32 bits (4 byte samples) with 3 channels of 1000 samples per second. Either reduce the number of channels to two or set 24 bits (3 bytes per sample) at this sample rate

Switching to the HEADER will confirm that the selection has been accepted.
When the selection is as required the parameters can be saved to the digitizer with the WRITE button on the CONTROL page.

Please note that this process takes at least 30 seconds, so under no circumstances should the power be removed from the digitizer for approximately 1 minute, to ensure that the process has been completed.

3 Installation And Operating Instructions

The digitizer is a sealed unit but it is recommended that it be protected from the weather. This will keep the unit clean and pleasant to handle and guard against the ingress of dirt and water whenever the connectors are removed.

The maximum operating temperature of the unit is +50 °C. It is possible for the unit to easily reach this temperature in moderate climates if it is left in the sun. The minimum operational temperature is -20 °C. The units will operate for extended periods at these temperatures however, it is obviously better if the extremes can be reduced.

3.1 Making The External Connections

Refer to section 6 for the specific model connectors and their pin allocations.

A nominal 12-volt supply (12 to 28 volts) should be connected to the designated pins on the POWER or LINE connector. The supply should ideally be current limited to no more than 1.5A to avoid problems arising from accidental shorts on the power supply lines. If this same supply is to be used for powering the sensors pay careful attention to ground loops. There should be no direct ground connection between the sensors and the SENSOR connectors GND pin in this situation because the ground reference is already established. Connect "signal-" to the sensor ground and "signal+" to the sensor output if the sensor does not have differential outputs.

Reverse supply connection will result in non-operation, but will not damage the unit.

The SERIAL or LINE connector provides output data in both RS232 and RS485. The connection should be made to the device receiving the digitizer data. If RS232 is used then cable length can be up to 50ft for baud rates up to 19200, reducing to 10ft at 115,200. If RS485 is used then data from the digitizer can be sent up to 4000ft at any baud rate if the line is properly terminated with a 50-ohm resistor at the receiving end. It is important to bear in mind that using RS485 with 50-Ohm termination at the receiving end will increase digitizer current consumption by 40mA. Using a 1-kOhm termination should allow 100ft cable lengths with minimal power consumption increase. CTS and CD are available on the SERIAL MIL-C connector only, to assist operation with Bi-directional Modems. See section 6 for connection details. (Not available on early versions).

If the low-resolution inputs are to be used great care must be taken to avoid ground loops. This is especially true because these are not balanced inputs and a ground connection will be necessary between the digitizer and the signal source. Any connection, which causes ground current through the SENSOR connectors, could potentially cause interference on the main channels. A 47R resistor in the ground wire will drastically reduce any interference and cause only a small degradation in the accuracy of the low-resolution channels. The sensitivity of the low-resolution channels is ±5volts and has a resolution of 12 bits.

The sensors should be connected to the digitizer using multi-way screened cable, ideally with individual twisted pairs. It is not normally necessary to have individually screened pairs of conductors for each channel unless the cable is in excess of 10 metres or the higher sample rates are being used.

The screen should be connected at one end only, preferably at the sensor end to avoid any ground current along the screen. A ground connection between the sensors and the digitizer is only required when no other ground connection between them exists. This will be the case if the sensors are passive, are active but powered from a totally separate supply e.g., solar panel and battery, or are inherently isolated from their power source by virtue of their construction. Any required ground connection between the sensor and the digitizer should be made using a conductor within the cable and not by using the screen.

Signal connection should be differential wherever possible. Even when the sensor output is unbalanced both “signal+” and “signal-” should be taken to the sensor and connected “signal-” to sensor gnd and “signal+” to sensor output at the sensor end.

The GPS antenna should be sited for maximum view of the sky and plugged into the GPS socket at the rear of the digitizer. The internal GPS unit is a Trimble Lassen iQ, and the antenna connector is a TNC type. The GPS antenna cable is normally 15 metres long, but can be supplied in shorter lengths if required.

The digitizer panel connectors are provided with dust caps. It is recommended that any connector not in use should have its appropriate dust cap attached.

4 Detailed Configuration

4.1 Overview

The configuration of the unit may either be performed using a web browser or configured by editing a simple configuration file. The file contains a list of ASCII entries, which can easily changed with any simple test editor.

This section contains a list of the entries that customise the digitizer parameters controlling such items as sample rate, number of channel, output data format etc.

The configuration file resided on one of the onboard flash devices which is itself configured as a regular disk device **/dev/mtdblock1** which is mounted as device **/mnt/mtda**.

The file is accessed by the unit, telnet or ftp as **/mnt/mtda/users/config.ini**.
Each line is a self contained entity describing a single configuration parameter.

The file can be edited with any text editor.

The configuration file is not case sensitive, but if editing externally, transfer it as ASCII when using FTP.

Spaces are allowed before or after the equal (=) operator on any entry.

Blank lines are allowed.

Separate multiple values in an entry line with a comma.

Any modification to the configuration file over the Ethernet connection will cause the digitizer acquisition to restart and action any changes.

Where an entry is to be defined as true (or on) it is normal to define the entry as **item_name**=1. In actual fact the value to the right of the name can be any value other than 0 (zero). Defining **item_name**=0 is the same as not defining it at all.

All inifile entries are vetted at run time. Any line that is not a comment or blank is verified for good syntax and where values must lie within specific bounds the content of the line is also tested.

Two actions are taken whenever a bad line is encountered.

1. The line will be ignored and the default value for the entry will be taken.

2. An entry is made in the message log to the effect
   "WARNING: Inifile error line nnn "entry name in error"

Entries that refer to channels have a channel number **xx** where **xx** is the one or two digit channel number.

Take an example of "**channel_0_samplerate=100**". The item to the left of the equal size describes the entry and to the right describes a value or directive. In this case channel 0 is set to 100 samples per second.

4.2 The Configuration File "config.ini"

4.2.1 Sections

All of the following entries live in a declared section called either **[recorder] or [timer]**. These are the only mandatory entry's and must occupy its own line and precede all of the entries belonging to that section. A section name splits the inifile into logical sections. A section name is a text item enclosed within square brackets. There are two sections currently defined which are *[recorder]* and *[timer]*. Each entry must be placed in its correct section as shown before the entry description.

4.2.2 Comments

Entries beginning with a comma (;) causes the entire line to be treated as a comment such as; This **line is a comment.**

A semicolon on the same line as an entry is allowed providing there is at least one space after the last character of the entry.

4.2.3 Sample Rates

[recorder]
channel_xx_samplerate=**200**

This entry sets the sample rate in samples per second for channel xx. A value of zero switches a channel off. The default is channels 0,1 and 2 recording at 100 s/s and all other channels off.
Channels 0 to 5 inclusive are the primary data channels.
Channels 6 to 11 inclusive are the secondary sample rate channels.
Channel 12 is the GPS channel and should never have an entry.
Channels 13 to 29 are the auxiliary status channels. A value other than 0 causes the status to be sampled at 1 sample per second irrespective of the value.

The sample rate can be set on a channel-by-channel basis. Channels 0 to 5 are
For the primary channels the available sample rates are: -

1, 2, 4, 5, 10, 20, 25, 40, 50, 75, 100, 120, 125, 150, 200, 250, 300,
375, 500, 600, 750, 1000, 2000, 3000

4.2.4 Channel Gain

[recorder]
channel_xx_normal_gain=**n**
channel_xx_high_gain=**n**
channel_xx_very_low_gain=**n**

One of three gain ranges may be selected for a set of three/six channels. With xx = 0 the gains for channels 0,1 & 2 will be set. With xx = 3 channels 3,4 & 5 will be set. Having entries with xx = 1,2,4 or 5 will have no effect.

Normal gain gives an input sensitivity of 1 micro-volt per bit on the 24th bit of a sample.
High gain gives an input sensitivity of 100 nano-volts per bit on the 24th bit of a sample.
Very low gain gives an input sensitivity of 2.5 micro-volts per bit on the 24th bit of a sample.

Omitting these entries causes normal gain to be selected.

4.2.5 Sample Size

[recorder]
bytes_per_sample=4

The bytes per sample entry define how many bytes (2=16bits, 3=24bits, 4=32bits) are stored for each primary channel. The input converters always operate on 32 bit data but the subsequent compression will remove the least significant 8 bits or least significant 16 bits of the conversion. This saves on storage by providing better compression.

4.2.6 Channel Names

[recorder]
channel_xx_long_id=**primary channel 0**

This entry defines the channel stream name associated with physical channel **xx**.
The above example will generate data files like: -
/data/113/edatayymmddhhmmss.primary **channel 0**

[recorder]
channel_xx_short_id=**p0**

This entry sets the channel identifier in the fixed data header of a miniseed file. Only the first three characters are used, hence you are given the option of placing a shortened channel name. If omitted the first three characters from channel_xx_long_id are used.

4.2.7 Channel Usage

There are 32 data channels each of which has a predetermined use, which cannot be changed. The following table lists each channel together with what it is used for.

Channels 0 to 5 are the primary input channels.
Channels 6 to 11 are secondary sampling channels.
Channel 12 is the GPS data channel
Channels 13 to 31 are auxiliary data channels with various functions as described.

Channel	Usage	Long name	Short name
0	Primary 0	pri0	p0
1	Primary 1	pri1	p1
2	Primary 2	pri2	p2
3	Primary 3	pri3	p3
4	Primary 4	pri4	p4
5	Primary 5	pri5	p5

6	Secondary 0	sec0	s0
7	Secondary 1	sec1	s1
8	Secondary 2	sec2	s2
9	Secondary 3	sec3	s3
10	Secondary 4	sec4	s4
11	Secondary 5	sec5	s5
12	GPS channel	gps	gps
13	Ambient temperature	aux0	a0
14	Not used	aux1	a1
15	Supply Current	aux2	a2
16	Supply Voltage	aux3	a3
17	Not used	aux4	a4
18	Analogue 0 input 0	aux5	a5
19	Analogue 1 input 1	aux6	a6
20	Analogue 2 input 2	aux7	a7
21	Analogue 3 input 3	aux8	a8
22	Module 0 temperature	aux9	a9
23	Module 1 temperature	aux10	a10
24	Module 2 temperature	aux11	a11
25	Module 3 temperature	aux12	a12
26	Module 4 temperature	aux13	a13
27	Module 5 temperature	aux14	a14
28	Not available	aux15	a15
29	Not available	aux16	a16
30	Not available	aux17	a17
31	Not available	aux18	a18

4.2.8 Channel Formats

[recorder]
channel_xx_format=**xyz**

This entry selects which format data is saved in.
xyz can be “mini_seed” “ascii” “gps” “gps2” “gps3” and “gps4”

4.2.8.1 Primary Data Channels

Channels 0 to 5 inclusive are **“mini_seed”**.
The available formats for these channels are **“mini_seed”** and **“ASCII”**
The format **“mini_seed”** is defined by the standard.
The format **“ASCII”** is a newline delimited data series in micro-volts.

4.2.8.2 Secondary Data Channels

Channels 6 to 11 inclusive are “Miniseed”.
The available formats for these channels are the same as for the primary channels.

4.2.8.3 GPS Channel

Channel 12 is “gps”.
The available formats for this channel are “gps”, ”gps2”, ”gps3” and “gps4”

gps - records the complete GPS message, no **pll data,** no **status data**.
gps2 – records **date**, **time**, **age** of **pll data**, **pll data** once per second.
records GPS **lock** and **antenna** status once per second.
gps3 - records whenever **pll age** or **pll data** changes.
records GPS lock and antenna status whenever it changes.
gps4 – records whenever **pll data** changes and when lock is resumed but
is suppressed when the **pll data** is within 10 micro seconds of GPS.
records GPS lock and antenna status whenever it changes.

The object is to record accurate timing information whilst conserving disk space.
Irrespective of which option is selected at least one complete GPS message is saved in each day directory.

4.2.8.4 Auxiliary Channels

Channels 13 to 31 are always “ASCII”.
There is no alternative for these channels.

4.2.9 USB Interface

[recorder]
usb=**xyz**

The usb interface is provided with a power saving mode. Typically the usb consumes 300mW plus power for any device. The interface may be configured to be “powered on demand” whenever any storage device is storing mini seed data.

"enabled" is the power on demand mode and is the default mode.
“disabled" and the usb interface will never be powered.
"always" and power will always be supplied to the interface.

4.2.10 Serial Data Output

[recorder]
serial_data=**xyz**

Three options control the serial output from the digitizer.

"legacy" Will send data out in the propriety legacy "Earth Data" format. This format consists of a number of headers followed by channel data series in a multiplexed format. This form of data supports one to six channels but all channels must have the same sample rate. Data are stored in their raw form, no compression takes place at all.

"compressed" will send data out in propriety "Earth Data" compressed format. This is a highly compressed form where each channel can have different sample rates.

"none" disables all serial data output.

4.2.11 Serial Baud Rate

[recorder]
serial_baud=**xyz**

Supported baud rates are 2400 4800 9600 19200 38400 57600 115200 230400.

4.2.12 Ethernet Interface

[recorder]
ethernet=**xyz**

The ethernet interface is provided with a power saving mode. Typically the ethernet consumes 300mW. The interface may be configured to be "powered on demand" and will be enabled whenever an Ethernet connection is present.

"auto" This is the default mode
"on" The Ethernet interface is always powered.
"off" The Ethernet interface is always unpowered.

4.2.13 Ethernet Proprietary Data

[recorder]
socket_data=**xyz**

Two Earth Data propriety formats are available from the Ethernet.

"legacy" Will send data out in the propriety legacy "Earth Data" format. This format consists of a number of headers followed by channel data series in a multiplexed format. This form of data supports one to six channels but all channels must have the same sample rate. Data are stored in their raw form, no compression takes place at all.

"compressed" will send data out in propriety "Earth Data" compressed format. This is a highly compressed form where each channel can have different sample rates.

socket_port=**xyz**
This is the decimal port number used to transport the data over a TCP/IP link. This can be any number between 256 and 65534 and will need to be agreed with your network administrator. A value of 30000 is normally a safe bet.

4.2.14 Ethernet Seedlink Data

The unit will send SEEDLINK protocol data as produced and maintained by GFZ at Potsdam University in Berlin. The digitizer with optional inbuilt disk storage is capable of supporting a SEEDLINK server transmitting data over Ethernet TCP/IP. Refer to http://www.gfz-potsdam.de/geofon/seiscomp/seedlink.html for further information on the SEEDLINK protocol itself.

The Seedlink system requires a file structured device to host its ring buffer which will provide about 30 days of history data. Three entries are required to enable Seedlink data.

[recorder]
seedlink=1

This entry allows the Seedlink system to start.

[recorder]
seedlink_storage=xyz

This entry defines the location of the seedlink data buffer. The two allowable devices are the internal compact flash (seenlink_storage=INTCF) or a mass storage system on the USB bus (seedlink_storage=OTHERUSB).
If USB storage is selected the USB interface must be continuously powered, refer to section 4.2.9

4.2.15 Mseed Master Storage

[recorder]
master_storage=**xyz**

Mini-Seed is the main data format the unit supports. This entry defines the storage medium. The two allowable devices are the internal compact flash (seenlink_storage=INTCF) or a mass storage system on the USB bus (seedlink_storage=OTHERUSB).
If USB storage is selected the USB interface must be enabled or continuously powered, refer to section 4.2.9

4.2.16 Mseed Secondary Storage

[recorder]
secondary_storage=**xyz**

Not currently supported

4.2.17 Local Volatile Storage

As incoming data is collected it is compressed and held in local onboard DRAM. This data is volatile and will be lost when power is removed. Two entries are provided to control the behaviour regarding when data is flushed to permanent storage (USB or Compact Flash). The USB interface has a power implication in that an extra 300mW is consumed whenever it is powered. The strategy used is to save data in DRAM for a period of time then flush it to permanent storage from time to time. The shorter it is in DRAM the more times the USB needs to be powered, the longer it is in DRAM increases data loss in the event of a power failure.

[recorder]
need_storage_trip=**xyz**

This entry defines as a percentage of the available volatile DRAM needs to be filled before data is committed to a USB device or Compact Flash. This defaults to 50% but could be reduced or increased. There is no hard and fast rule but as a guidance do not reduce it below 20% or increase it above 80%.

[recorder]
longflush_timeout=**xyz**

The green flush button on the unit powers the USB and commits any outstanding data to it. Pressing the button for about 5 seconds will keep the USB power on for this period in minutes. This is really only useful when there is say a Bluetooth USB dongle or Wi-Fi device and permits continuous connection with the unit for an extended period of time.

4.2.18 Mseed Endian

[recorder]
endian=**big**
endian=**little**

This entry specifies the endian order of data in a miniseed file. The default is big endian which is the endian system used by Motorola and Sun etc. Endian “little” is the system used in the PC world.

4.2.19 Mseed File Sizes

[recorder]
mseed_filesize=**xyz**

This entry sets the file size in minutes before a new file is created. Allowable file lengths are 0,1,2,5,10,20,30 and 60 minutes
Defining this to a length of other than 0(zero) is enough to enable the Mseed system.

4.2.20 Mseed Header Fields

[recorder]
station_short_identifier=**xyz**

The first 5 characters of this entry are places in the station field in the Mseed header.

[recorder]
location_identifier=**xyz**

The first 2 characters of this entry are places in the location field in the Mseed header.

[recorder]
network_code=**xyz**

The first 2 characters of this entry are places in the network field in the Mseed header.

4.2.21 Initial Position

[recorder]
initial_position=**xxx,lat.lon,alt**

This entry is a helper for the GPS positioning system which may speed up initial GPS lock by providing the approximate co-ordinates of the unit. The three parameters are latitude,

longitude in degrees and altitude in metres. Separate each argument with a comma. Each argument is of type float (50.5 is 50 degrees 30 minutes).

4.2.22 GPS Command

[recorder]
gps_command=**any ASCII text string**

Not currently supported.

4.2.23 GPS Power Cycling

[recorder]
gps_low_power=**cycle_time,on_time**

This entry power cycles the GPS unit. Both cycle_time and on_time are specified in seconds. So, for example gps_low_power=3600,300 powers the GPS on for 5 minutes (300 seconds) once every hour (3600 seconds). The default when not specified is to have the GPS always powered. When in power cycle mode the on period cannot be less than 5 minutes (300 seconds) and more than 16.65 minutes (999 seconds). The off period cannot be less than 5 minutes and more than 8 hours (28800 seconds).

4.2.24 External GPS

[recorder]
external_gps=1

Defining this will cause the internal GPS to be not used and the optional external GPS will be used for unit timing.

4.2.25 GPS Power Saving

[recorder]
gps_low_power=**cycle_time,on_time**

This entry power cycles the GPS unit. Both cycle_time and on_time are specified in seconds. So for example gps_low_power=3600,300 powers the GPS on for 5 minutes (300 seconds) once every hour (3600 seconds). The default when not specified is to have the GPS always powered. When in power cycle mode the on period cannot be less than 5 minutes (300 seconds) and more than 16.65 minutes (999 seconds). The off period cannot be less than 5 minutes and more than 8 hours (28800 seconds).

4.2.26 Digital Outputs

[recorder]
cald0=**ddmmyyyy, hhmmss, duration, repeat, type**
cald1=**ddmmyyyy, hhmmss, duration, repeat, type**

These entries define the state of the digital output lines d0 and d1 respectively. The d0 output is an open collector driven output and d1 is a CMOS driven output. The outputs are asserted (active) or not asserted (at rest).

The output is asserted at the time and date specified for the **duration** in seconds. The **type** specifies the state when asserted and at rest**. type** has two states as shown below: -

- d0 Ouput
  - **type=0**
    Rest state is high impedance, single pulse of low impedance for **duration.**
  - **type=1**
  - Rest state is low impedance, single pulse of high impedance for **duration.**
- d1 Output
  - **type=0**
    Rest state is low CMOS voltage, single pulse of high voltage for **duration.**
  - **type=1**
    Rest state is high CMOS voltage, single pulse of low voltage for **duration.**

- **repeat**
  Allows the calibrate sequence to be repeated every **repeat** minutes.

The date and time specified must be in this century.

4.2.27 Sensor Calibration

These entries governs the enabling and sending of a calibration signal to the sensor. The Enable is achieved via one of the digital outputs dig0 (the open collector) or dig1 (the TTL output) being driven high or low. For example, to enable using the open collector output being driven low, set cal_enable=dig0lo. The calibration enable duration (cal_enable_duration) is set to the number of seconds required. The calibration signal type (cal_type) can be a sine wave (sine), square wave (pulse) or white noise (noise). The amplitude of the calibration signal (cal_amplitude) is set to a number of volts and the frequency of the signal (cal_frequency) set to Hertz. Should the calibration signal require to be sent periodically then set the cal_repeat to begin at ddmmyyyy, hhnnss and reoccur every 'repeat' minutes.

[recorder]
cal_enable=[dig0hi][dig0lo][dig1hi][dig1lo]
cal_enable_duration=seconds // 0=disable or 1 to 86400 seconds
cal_type=[sine][pulse][noise]
cal_amplitude=[volts]
cal_frequency=[frequency]
cal_repeat=ddmmyyyy, hhnnss, repeat

For legacy compatibility, the following is still supported.

[recorder]
cala= **ddmmyyyy, hhmmss, duration, repeat, type, amplitude1, freq1, freq2**

This entry defines the state of the analogue output line. This output is an analogue driven output of amplitude **amplitude1**. The amplitude is specified in millivolts in the range +/- 10000 millivolts. The output is asserted (active) or not asserted (at rest). The output is asserted at the time and date specified for the **duration** in seconds. The **type** specifies the state when asserted and at rest**. type** has three states as shown below:-
type=0 Rest state is 0 volts, single pulse of amplitude1 for **duration**
type=1 Rest state is 0 volts, sine wave of amplitude1 for **duration**
type=2 Rest state is 0 volts, swept sine **freq1** to **freq2** over **duration**
repeat allows the calibrate sequence to be repeated every **repeat** minutes.
The swept sine calibration has two additional fields. **freq1** is the start frequency in Hz and **freq2** is the end frequency in hertz. A sine wave is swept between the two frequencies over a period of **duration**.
The date and time specified must be in this century.

4.2.28 Sensor Mass Positioning

These entries govern the automatic mass positioning of the connected sensor. The Enable is achieved via one of the digital outputs dig0 (the open collector) or dig1 (the TTL output) being driven high or low. For example, to enable using the open collector output being driven low, set autz_enable=dig0lo. The mass positioning enable duration (autz_enable_duration) is set to the number of seconds required.

The allocation of auxiliary channels (0…7) to monitor the mass positions for the Vertical (Z/W), North/South (Y/V) and East/West (X/U) axes are set using the parameters: 'autz_vert_aux', 'autz_ns_aux' and 'autz_ew_aux' respectively. Setting of the minimum and maximum monitored voltages (for the respective axes) outside of which auto mass positioning will be triggered is achieved by setting 'autz_vert_low', 'autz_vert_high', 'autz_ns_low', 'autz_ns_high', 'autz_ew_low' and 'autz_ew_high' to the desired voltages.

[recorder]
autz_enable=[dig0hi][dig0lo][dig1hi][dig1lo]

autz_enable_duration=[seconds] // 0=disable or 1 to 60
autz_vert_aux=[channel] // [0 to 7]
autz_ns_aux=[channel] // [0 to 7]
autz_ew_aux=[channel] // [0 to 7]

autz_vert_low=[voltage] // [0.1 to 9.99]
autz_ns_low=[voltage] // [0.1 to 9.99]
autz_ew_low=[voltage] // [0.1 to 9.99]
autz_vert_high=[voltage] // [0.1 to 9.99]
autz_ns_high=[voltage] // [0.1 to 9.99]
autz_ew_high=[voltage] // [0.1 to 9.99]

4.2.29 Burst Mode Recording

[recorder]
burst= **ddmmyyyy, hhmmss, duration, repeat**

This entry defines that the primary data channels are to operate in burst mode. Recording on the primary channels will commence at the time and date specified for the **duration** in minutes. The argument **repeat** allows the burst sequence to be repeated every **repeat** minutes. The date and time specified must be in this century. Restrictions apply to burst mode recording, refer to section 1.13 for details of these restrictions.

4.2.30 Window Timer

[timer]
tw=**ddmmyyyy hhmm duration repeat**

This entry defines a timer window. A timer entry is a list of one or more lines describing the recording pattern for the unit. In the absence of any entries the recorder records continuously. The value **repeat** is optional. Up to 200 entries can be made at any one time, all specifying a different start date and duration. A single entry can be made with a **repeat** field that will cause the recorder to record data at a specific time each day or week. The fields duration and **repeat** are specified in minutes. For example the entry *tm=01082002 1400 60 480* will cause 60 minutes to be recorded every 480 minutes (8

hours). That is to say three times per day at 14:00, 22:00 and 06:00. If the date and time specified by *ddmmyyyy hhmm* is in the future the recorder will stay off until that future date and time. As that date and time becomes history the **repeat** value is used to calculate either the end of a current session or the beginning of the next session. Note.. Currently the timing windows are not implemented.

4.2.31 Log File Size

[recorder]
messagelog_size=**size**

This entry limits the size of the recorder log file. The value **size** is specified in kbytes. For example a value of 10000 limits the log size to 10000*1024 bytes, which is just over 10Mbytes. The default size when not specified is 5Mbytes.

5 RS232 (RS485) Control and Data Formats

5.1 Description

Both RS232 and RS485 data outputs are available at the same time. Data is transmitted in one-second packets. Control is only via RS232.

This section details operation via RS232 because this is easily implemented using a standard PC.

Connecting the RS232 to a PC establishes a two-way communication between the digitizer and the PC. Control information can be **received** from the PC and data and status information **transmitted** from the digitizer.

5.2 Connection

The simplest way to confirm correct connection is to install MODMON32.
The time series should be displayed on the CHANNEL page 10 to 15 seconds after powering the digitizer. Confirm that changes can be made to the operating mode using the CONTROL and HEADER pages. As long as the WRITE button is not pressed turning off the power to the digitizer will clear any changes made. Next time it is powered the original parameters will be used. An alternative is to use HyperTerminal at 19200 Baud, 8 Bits, no Parity, 1 Stop bit, by this method the digitizer initial self test sequence can be viewed but the MODMON32 data which follows will not be discernable.

5.3 Transmit Packet Format From The Digitizer

Serial data can be transmitted by the Digitizer in 2 modes of operation.

- Legacy Mode
  (Compatible with legacy Earth Data Digitizer MODMON output)

- Compressed Mode

The Data can be sent via the serial RS232/485 (default 19200 Baud, 8 Bits, no Parity, 1 Stop bit) or Ethernet transports (TCP/IP default port 30000).

5.3.1 Legacy Mode

The digitizer transmits a header and a block of data once every second irrespective of the selected configuration or baud rate. The format of the header is shown below and follows a convention common to headers produced by other Earth Data equipment in that the first two entries depict the header type and the remaining header length.

5.3.1.1 MOD Header

Byte	Type	Size In Bytes	Name	Function
0	char	4	MOD Header Id	Identifier “MOD<0>“ ASCII
4	long	4	Size of MOD	184 = size of MOD in bytes (excl. Identifier and size)
8	char	12	Device ID	ASCII
20	char	6	Version	ASCII
26	char	1	Space	ASCII space
27	char	4	serial No.	ASCII SN
30	char	1	Space	ASCII space
32	char	8	test pattern	binary (hex 22,22,00,55,55,00,ff,ff)
40	long	4	block count	binary 32 bit block count
44	int	2	Ncomps	binary number of components in data
46	int	2	sample rate	samples per block per component
48	int	2	Bytesper	bytes per sample to send
50	char	2	filter type	“LP” linear phase “MP” minimal phase
52	int	2	deci12	decimation factor of filter 3
54	int	2	deci10	decimation factor of filter 4
56	int	2	Plldata	16 bit phase error in 1 second pll
58	char	1	Gain	"G" ASCII
59	char	1	Gain	Binary representation of channel gain.
60	long	4	Cal gain C1	Component 1 gain value for Hi/Lo gain
64	long	4	Cal gain C2	Component 2 gain value
68	long	4	Cal gain C3	Component 3 gain value
72	long	4	Offset C1	Component 1 offset value for Hi/lo gain
76	long	4	Offset C2	Component 2 offset value
80	long	4	Offset C3	Component 3 offset value
84	Int	2	ADC 0	Supply Voltage at Supply connector
86	Int	2	ADC 1	Supply current
88	Int	2	ADC 2	Temp. sensor voltage
90	Int	2	ADC 3	Supply Voltage at Remote connector
92	Int	2	ADC 4	User input
94	Int	2	ADC 5	User input
96	Int	2	ADC 6	User input
98	Int	2	ADC 7	Not used
100	char	2	coupling	"DC" ASCII
102	long	4	ltime	Time in 32 bit time_t format
106	long	4	reserved	
110	long	4	reserved	
114	Int	2	reserved	bit flags of GPS status
116	long	4	GPS block	block count of last valid GPS
120	char	72	GPS message	Last valid GPS message

Most of these are self-explanatory. The following may require further clarification.

Offset 40 block count: Increments every second starting from zero at switch on. It can be used to indicate the number of seconds the unit has been powered. When making long files with only one MOD header every file it can be used to check that the right number of packets are in each file. It is also useful when used in conjunction with Offset 116 for monitoring GPS lock. See below.

Offset 44 Ncomps: 1 to 6 channels (components) can be included in the data. The default is 3 components and can be changed by MODMON32 or other specifically written software. One component uses the vertical input. Two component uses vertical and north/south.

Offset 46 sample rate: Number of samples per second per component. The default is 100 samples/second.

Offset 48 bytesper: The digitizer can be programmed to send 3, or 4 bytes per sample. The default is 4 corresponding to 32 bits

Offset 50 filter type: Linear phase only is available. Minimal phase filtering will be available later. All channels must have the same filter type.

Offset 52, 54 deci12, deci10: This indicates the decimation being performed by the two internal filters. They will change according to the selected sample rate and cannot be directly programmed. The sample rate is achieved by dividing 3000 by the product of deci12 and deci10 which assume integer values between 1 and 12 and 1 and 10 respectively. Only those divisions producing sensible sample rates are used.

Offset 56 Plldata: This value indicates the error between the GPS one pulse per second and the internal one pulse per second. It is an integer value indicating the error in multiples

of 1µS. This should normally fluctuate between +5 and –6 indicating an error between +5 and -6µS.

Offset 58, 59 gain: Four pre-amplifier gain settings are available.

- **Low** Gain (Lo)

  Provides 1µV per bit (24 bit word size)
- **High** Gain (Hi)
  Provides 100nV per bit (10x Lo)
- **Very Low** Gain (VLo)

  Provides 2.5µV per bit (0.4x Lo)
- **Very High** (VHi)
  Provides 20nV per bit (50x Lo)

Offset 58 contains ASCII “G” (0x47)

The second byte (offset 59) will be set differently according to the number of channels.

Number Channels 1 - 3

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	1	1	0	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, Vlo 1 = Hi, V Hi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 1.1
Channels 1,2,3 ‘Lo’ gain = 0x30

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	0	1	1	0	0	0	0
0	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	1	1	0	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, Vlo 1 = Hi, V Hi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 1.2
Channels 1,2,3 ‘Hi’ gain = 0x37

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	0	1	1	0	1	1	1
0	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	1	1	0	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, Vlo 1 = Hi, V Hi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 1.3
Channels 1,2,3 ‘V Lo’ gain = 0x70

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	1	1	1	0	0	0	0
0	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	1	1	0	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, Vlo 1 = Hi, V Hi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 1.4
Channels 1,2,3 ‘V Hi’ gain = 0x77

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	1	1	1	0	1	1	1
0	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	1	1	0	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, Vlo 1 = Hi, V Hi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Number Channels 4 – 6

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0 = Lo, Hi 1 = VLo, VHi ch: 4,5,6	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	0 = Lo, VLo 1 = Hi, VHi ch 6	0 = Lo, VLo 1 = Hi, VHi ch 5	0 = Lo, VLo 1 = Hi, VHi ch 4	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, VLo 1 = Hi, VHi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 2.1
Channels 1,2,3,4,5,6 'Lo' gain = 0x00

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	0	0	0	0	0	0	0
0 = Lo, Hi 1 = VLo, VHi ch: 4,5,6	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	0 = Lo, VLo 1 = Hi, VHi ch 6	0 = Lo, VLo 1 = Hi, VHi ch 5	0 = Lo, VLo 1 = Hi, VHi ch 4	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, VLo 1 = Hi, VHi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 2.2
Channels 1,2,3,4,5,6 'Hi' gain = 0x3F

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	0	1	1	1	1	1	1
0 = Lo, Hi 1 = VLo, VHi ch: 4,5,6	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	0 = Lo, VLo 1 = Hi, VHi ch 6	0 = Lo, VLo 1 = Hi, VHi ch 5	0 = Lo, VLo 1 = Hi, VHi ch 4	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, VLo 1 = Hi, VHi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 2.3
Channels 1,2,3,4,5,6 'VLo' gain = 0xC0

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
1	1	0	0	0	0	0	0
0 = Lo, Hi 1 = VLo, VHi ch: 4,5,6	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	0 = Lo, VLo 1 = Hi, VHi ch 6	0 = Lo, VLo 1 = Hi, VHi ch 5	0 = Lo, VLo 1 = Hi, VHi ch 4	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, VLo 1 = Hi, VHi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 2.4
Channels 1,2,3,4,5,6 'VHi' gain = 0xFF

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
1	1	1	1	1	1	1	1
0 = Lo, Hi 1 = VLo, VHi ch: 4,5,6	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	0 = Lo, VLo 1 = Hi, VHi ch 6	0 = Lo, VLo 1 = Hi, VHi ch 5	0 = Lo, VLo 1 = Hi, VHi ch 4	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, VLo 1 = Hi, VHi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 2.5
Channels 1,2,3 'Lo' gain, Channels 4,5,6 'Hi' gain = 0x38

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	0	1	1	1	0	0	0
0 = Lo, Hi 1 = VLo, VHi ch: 4,5,6	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	0 = Lo, VLo 1 = Hi, VHi ch 6	0 = Lo, VLo 1 = Hi, VHi ch 5	0 = Lo, VLo 1 = Hi, VHi ch 4	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, VLo 1 = Hi, VHi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Example 2.6
Channels 1,2,3 'VHi' gain, Channels 4,5,6 'VLo' gain = 0xC7

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
1	1	0	0	0	1	1	1
0 = Lo, Hi 1 = VLo, VHi ch: 4,5,6	0 = Lo, Hi 1 = VLo, VHi ch: 1,2,3	0 = Lo, VLo 1 = Hi, VHi ch 6	0 = Lo, VLo 1 = Hi, VHi ch 5	0 = Lo, VLo 1 = Hi, VHi ch 4	0 = Lo, VLo 1 = Hi, VHi ch 3	0 = Lo, VLo 1 = Hi, VHi ch 2	0 = Lo, VLo 1 = Hi, VHi ch 1

Offset 60 to 68 Cal gain: The digitizer gain can be calibrated against its own internal 5 volt reference. This is entirely separate from the reference used in the A to D circuitry. MODMON32 or specifically written software can invoke this gain (and offset) calibration at any time without the need for any additional external connections or the need to remove any input signals. If required the calibration values can be written permanently to the digitizer. These locations store the multiplier applied to each data sample to produce correct gain. Every time an auto-calibration is performed these values are updated.

Offset 72 to 80 Offset: These locations store the offsets values, which are added to (or subtracted from) every data sample such that the digitizer introduces very small d.c. offsets. Every time the unit is switched the offsets are calibrated and these values amended. This will not, however, remove any d.c. offset from the signal itself. It is purely a calibration function.

Offset 84 to 99 ADC values: The digitizer produces 8 low resolution ADC values every second. Of these, three are available to the user, The usable range is -5 volts to +5 volts. Other ADC values are internally monitored parameters as follows:

ADC0	Supply voltage at input Voltage = (ADC0*10.9/2700)+5
ADC1	Supply current. Current (Amps) = ADC1*22/170
ADC2	Internal temperature Temperature (°Celsius) = ADC2/10-50
ADC3	Supply voltage at remote input. Voltage = (ADC3*10.9/2700)+5
ADC4	User Input: Input voltage (volts) = ADC4/1000
ADC5	User Input: Input voltage (volts) = ADC5/1000
ADC6	User Input: Input voltage (volts) = ADC6/1000
ADC7	User Input: Input voltage (volts) = ADC6/1000

Offset 100 Coupling: Coupling is d.c.

Offset 102 to 105: This location contains the time of the packet as a second count starting on 1st Jan 1970. This is the original IBM-PC time code known variously as ctime, ltime and time_t. This is an ongoing count in seconds from 1st Jan 1970. This time is the internal digitizer time either locked to the GPS time or digitizer tracking time when the GPS is absent for periods.

Offset 106 to 113: Not used.

Offset 114 GPSflags: Not used.

Offset 116 GPSblock: This is the last value of block count (Offset 40) when the GPS (if fitted) produced an in-lock message. If the GPS is in lock the value will be the same as the block count. If a GPS has never been in lock (or is not fitted) this value will be zero. In general, this location can be used to determine how long ago the GPS lost lock.

Offset 120 GPS message: This message is the last valid message sent by the GPS unit (if fitted).
A typical message is as follows:

$GPRMC,020000,A,5055.8677,N,00130.0571,W,002.0,159.9,210998,004.4,W*73

The message can be broken down into sections as follows:

$GPRMC,	Type of message (Recommended Minimum specific NMEA)
020000,	Time hhmmss
A,	A=In lock: V=out of lock
5055.8677,N,	001°30.0571' West
00130.0571,W,	50° 55.8677' North
002.0,	Velocity over ground
159.9,	Heading
210998,	Date
004.4,	Magnetic variation
W	Direction of magnetic variation
*73	Message checksum

5.3.1.2 MDE Header

The Modulator Enhanced segment format data immediately follows the "MOD" header in Enhanced Mode.

Byte	Type	Size In Bytes	Name	Function
0	char	4	MDE Header Id	Identifier "MDE<0>" ASCII
4	long	4	Size of MDE	180 = size of MDE in bytes (excl. Identifier and size)
8	char	8	Enhanced Serial No.	ASCII SN
16	Int	2	deci10-5	decimation factor of filter 5
18	Int	2	deci10-6	decimation factor of filter 6
20	Int	2	deci10-7	decimation factor of filter 7
22	long	4	Cal gain C4	Component 4 gain value for Hi/Lo gain
26	long	4	Cal gain C5	Component 5 gain value
30	long	4	Cal gain C6	Component 6 gain value
34	long	4	Offset C4	Component 4 offset value for Hi/lo gain
38	long	4	Offset C5	Component 5 offset value
42	long	4	Offset C6	Component 6 offset value
46	Int	2	ADC 8	Input Module 1 Temp sensor
48	Int	2	ADC 9	Input Module 2 Temp sensor
50	Int	2	ADC 10	Input Module 3 Temp sensor
52	Int	2	ADC 11	Input Module 4 Temp sensor
54	Int	2	ADC 12	Input Module 5 Temp sensor
56	Int	2	ADC 13	Input Module 6 Temp sensor
58	Char	129	GPS message (Trimble)	Last valid GPS message

Offset 8 Enhanced serial number: Full representation of the unit serial number.

Offset 16,18,20 deci10-5,dec10-6,dec10-7: Reserved for additional sample rates.

Offset 22 to 30 Cal gain: Gain calibration.

Offset 34 to 42 Offset: These locations store the offsets values, which are added to (or subtracted from) every data sample such that the digitizer introduces very small d.c. offsets. Every time the unit is switched the offsets are calibrated and these values amended. This will not, however, remove any d.c. offset from the signal itself. It is purely a calibration function.

Offset 46 to 56 Module temperatures: The temperatures measured at each of the channel modules.

Input Module 1 Temperature (°Celsius) = ADC8/10-50
Input Module 2 Temperature (°Celsius) = ADC9/10-50
Input Module 3 Temperature (°Celsius) = ADC10/10-50
Input Module 4 Temperature (°Celsius) = ADC11/10-50
Input Module 5 Temperature (°Celsius) = ADC12/10-50
Input Module 6 Temperature (°Celsius) = ADC13/10-50

Offset 58 GPS message: This message is the last valid message sent by the alternative Trimble GPS unit (if fitted). A typical message is as follows:

CRLF>RAL47973+00000+00002;*4C<CRLF>RPV47973+5093117-0015013100000002;*7E
CRLF>RSTaabcddefgg;*ck<CRLF>RTM131921184260220011300410000;*62<

Refer to Trimble TAIP specification for full explanation.

5.3.1.3 Data Segment

The data immediately follows the "MOD" header, or the "MDE" header if enabled.
Channels can be switched on and off. This example is for three-channel operation.

Byte	Type	Size In Bytes	Name	Function
0	char	4	DAT Segment Id	Identifier "DAT<0>" ASCII
4	long	4	Size of DAT	size of DAT in bytes (Excludes Identifier and size)
8	int	n	Sample1 (from Channel 1)	
8+n	int	n	Sample1 (from Channel 2)	
8+2n	int	n	Sample1 (from Channel 3)	
8+3n	int	n	Sample2 (from Channel 1)	
8+4n	int	n	Sample2 (from Channel 2)	
8+5n	int	n	Sample2 (from Channel 3)	
8+6n	int	n	Etc	
8+7n	int	n	Etc	

Size "n" can be 3, or 4 depending on the number of bytes/sample selected. Default is 4 (32 bit).
Within a sample, least significant byte first, 2's compliment arithmetic.

This is an example of six-channel operation

Byte	Type	Size In Bytes	Name	Function
0	char	4	DAT Segment Id	Identifier “DAT<0>“ ASCII
4	long	4	Size of DAT	size of DAT in bytes (Excludes Identifier and size)
8	int	n	Sample1 (from Channel 1)	
8+n	int	n	Sample1 (from Channel 2)	
8+2n	int	n	Sample1 (from Channel 3)	
8+3n	int	n	Sample1 (from Channel 4)	
8+4n	int	n	Sample1 (from Channel 5)	
8+5n	int	n	Sample1 (from Channel 6)	
8+6n	int	n	Sample2 (from Channel 1)	
8+7n	int	n	Sample2 (from Channel 2)	
8+8n	int	n	Sample2 (from Channel 3)	
8+9n	int	n	Sample2 (from Channel 4)	
8+10n	int	n	Sample2 (from Channel 5)	
8+11n	int	n	Sample2 (from Channel 6)	
8+12n	int	n	Etc	
8+13n	int	n	Etc	

5.3.1.4 Checksum Segment

The checksum immediately follows the “DAT” segment.

Byte	Type	Size In Bytes	Name	Function
0	Char	4	SUM Segment Id	Identifier “SUM<0>“ ASCII
4	Long	4	Size of SUM	Size of this SUM section in bytes (Excludes Identifier and size)
8	Int	2	Reserved	Will always be zero
10	Char	1	ChecksumLo	Low 8 bits of 16 bit checksum
11	Char	1	ChecksumHi	High 8 bits of 16 bit checksum

The two-checksum bytes contain the sixteen bit arithmetic sum of all bytes in the MOD, MDE and DAT sections and the first ten bytes of the SUM section. In effect the checksum covers all data for a one-second period. If preferred the sum can be limited to eight bits and ChecksumHi can be discarded.

5.3.2 Compressed Mode

Each second's worth of data is transmitted in the following format, with header information followed by the compressed data for each channel, terminated with a CRC16 checksum. Each block of data reports its size independently. The Header is a fixed size, whereas the size of the data block will vary according to how compressed the data is.

Header		*Includes:* *Hardware Information* *Software Version* • • *Number of Channels* • *Timing Information* • *Positional Information* • *GPS Status Information* • *ADC (SOH) Channel Values*
Data (Channel 1)		*Includes:* *Channel Number (1)* *Sample Rate* • • *Bytes Per Sample* • *Gain Setting* • *Compression Info* • *Compressed Data* •
Data (Channel 2)		*Includes:* *Channel Number (2)* *Sample Rate* • • *Bytes Per Sample* • *Gain Setting* • *Compression Info* • *Compressed Data* •
…		
Data (Channel N)		*Includes:* *Channel Number (N)* *Sample Rate* • • *Bytes Per Sample* • *Gain Setting* • *Compression Info* • *Compressed Data* •
Checksum	CRC16	le_u_int16

EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

5.3.2.1 MO2 Header

char[4]	Id	MO2<null>
le_u_int16	Size	Header Size in bytes (excluding Id and Size), should be 108.
le_u_int16	Version	<hi nibble>.<lo nibble>
char	Device Id	0 = EDR-209 1 = EDR-210
char	Number of channels	1…12 (1…6 typically primary, 7…12 typically secondary)
le_u_int32	Serial Number	
le_u_int32	Seconds	UNIX epoch
le_u_int32	Last Valid GPS Block	UNIX epoch
le_u_int32	PLL Data	Phase Error
le_u_int32	Oldest Seconds	UNIX epoch (oldest available data in buffer)
le_u_int16	Year	
char	Month	1…12
char	Day	1…31
char	Hour	0…23
char	Minute	0…59
char	Second	0…59
GPS_Status *	GPS Status	(see below)
le_float32	Latitude	Radians North (positive) Radians South (negative)
le_float32	Longitude	Radians West (positive) Radians East (negative)
le_float32	Altitude	Metres
le_u_int32	ADC[16]	16 A to D Channels (SOH)

5.3.2.1.1 GPS Status

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
Usued	Usued	Usued	Usued	Battery	Antenna Fault Type	Antenna Fault	In Lock
				0 = Ok	If Antenna Fault = 1, then 0 = Open 1 = Short	0 = ok 1 = fault	0 = not in lock 1 = in lock

EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

5.3.2.2 Data Segment

char[4]	Id	DA2<null>
le_u_int16	DataSize	Size in bytes (excluding Id and DataSize)
le_u_int16	Sample Rate	Sample Count
char	Channel Number	0...11
char	Bytes Per Sample	1,2,3,4
char	Compression Info	Bits per Symbol
char	Gain	0 = Lo, 1= Hi, 2 = V Lo, 3 = V Hi
char[DataSize-6]	Data	Compressed Data

5.3.2.2.1 Compressed Data

If the [Compression Info] byte is zero (0) then the data will be uncompressed and be arranged as an array of [Sample Rate] bytes * [Bytes Per Sample] in (2's complement format). For example if Bytes Per Sample is 4 and the Sample Rate N sps, the Data will be arranged as N x.le_int32.

$Sample_1$	**$Sample_2$**	**$Sample_3$**	**…**	**$Sample_N$**
le_int32	le_int32	le_int32	…	le_int32

Alternatively, a non-zero value for the [Compression Info] (CI) indicates that the data is compressed and the adjacent sample differences are written in symbols, each comprising CI bits. Note that as the data is written as differences the first and last samples are written first, so that the differences can be extrapolated back to raw samples and the decompression checked.

Compressed Data						
$Sample_1$	**$Sample_N$**	**$Symbol_1$**	**$Symbol_2$**	**$Symbol_3$**	**…**	**Symbol…**
le_int32	le_int32	CI bits	CI bits	CI bits	…	CI bits

Each Symbol consists of CI bits, where

Bit_{CI}	**Bit_{CI-1}**	**Bit_{CI-2}**	**…**	**Bit_1**
Terminator			…	
1 = Terminating Symbol	Data			

If the Difference between adjacent samples fits into [CI-1] bits then the difference can be written in one symbol. However, if the data requires more bits, then additional symbol(s) will be required. The Symbol following a Terminating Symbol will be the first symbol for the next difference. The data is therefore arranged in consecutive symbols, each represented by CI bits, note:

$Difference_1 = Sample_2 - Sample_1$
$Difference_2 = Sample_3 - Sample_2$

EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

...

$Difference_{N-1} = Sample_N - Sample_{N-1}$

For example, consider the difference of +100, where the bits per symbol is set to 5.

+100 can be expressed in 2's complement format using 8 bits as:

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
0	1	1	0	0	1	0	0
Sign 0 = +tve 1 = -ve							

Therefore, using 5 bits per symbol, the difference requires 2 symbols.

i.e.

$Symbol_1$				
Bit_5	**Bit_4**	**Bit_3**	**Bit_2**	**Bit_1**
0	0*	1	1	0
Terminator	**Data**			

$Symbol_2$				
Bit_5	**Bit_4**	**Bit_3**	**Bit_2**	**Bit_1**
1**	0	1	0	0
Terminator	**Data**			

Note* **$Symbol_1$ *includes the sign information in the MSB of the Data*
(0 = + positive, 1 = - negative)
***Note* **$Symbol_2$** *includes the Terminator bit set, indicating it is the last symbol.*

The difference is computed from the 2 symbols as:

$Symbol_1$				**$Symbol_2$**			
Bit_4	**Bit_3**	**Bit_2**	**Bit_1**	**Bit_4**	**Bit_3**	**Bit_2**	**Bit_1**
0	1	1	0	0	1	0	0
2's compliment difference							

In order to represent this data as a 32 bit difference the MSB of the data is simply sign extended to the required 32 bits.

Similarly, -100 can be expressed in 2's complement format using 8 bits as:

Bit_8	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1
1	0	0	1	1	1	0	0
Sign 0 = +tve 1 = -ve							

Therefore, consider the case where 4 bit symbols are utilised; the difference requires 3 symbols.

i.e.

$Symbol_1$			
Bit_4	**Bit_3**	**Bit_2**	**Bit_1**
0	1*	1	0
Terminator	**Data**		

$Symbol_2$			
Bit_4	**Bit_3**	**Bit_2**	**Bit_1**
0	0	1	1
Terminator	**Data**		

$Symbol_3$			
Bit_4	**Bit_3**	**Bit_2**	**Bit_1**
1**	1	0	0
Terminator	**Data**		

Note* **$Symbol_1$ *includes the sign information in the MSB of the Data (0 = + positive, 1 = - negative)*
***Note* **$Symbol_3$** *includes the Terminator bit set, indicating it is the last symbol.*

The difference is computed from the 3 symbols as:

$Symbol_1$			**$Symbol_2$**			**$Symbol_3$**		
Bit_3	**Bit_2**	**Bit_1**	**Bit_3**	**Bit_2**	**Bit_1**	**Bit_3**	**Bit_2**	**Bit_1**
1	1	0	0	1	1	1	0	0
2's compliment difference								

In order to represent this data as a 32 bit difference the MSB of the data is simply sign extended to the required 32 bits.

Sample code for decompressing the data is included later.

EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

5.3.2.3 Checksum Segment

The CRC field is two bytes, containing a 16–bit binary value. The CRC value is appended to the message. The receiving device recalculates a CRC during receipt of the message, and compares the calculated value to the actual value it received in the CRC field. If the two values are not equal, there is an error in the data packet.

The CRC is started by first preloading a 16–bit register to all 1's. Then a process begins of applying successive 8–bit bytes of the message to the current contents of the register. Only the eight bits of data in each character are used for generating the CRC (for serial data, start and stop bits, and the parity bit, do not apply to the CRC).

During generation of the CRC, each 8–bit character is exclusive ORed with the register contents. Then the result is shifted in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position. The LSB is extracted and examined. If the LSB was a 1, the register is then exclusive ORed with a preset, fixed value. If the LSB was a 0, no exclusive OR takes place.

This process is repeated until eight shifts have been performed. After the last (eighth) shift, the next 8–bit byte is exclusive ORed with the register's current value, and the process repeats for eight more shifts as described above. The final contents of the register, after all the bytes of the message have been applied, is the CRC value. When the CRC is appended to the message, the low-order byte is appended first, followed by the high-order byte.

Sample CRC16 Calculation Code:

```
#define CRC16_POLYNOMIAL 0xA001          // Polynomial used for CRC-16 checksum

union CRC_doub
    {                                    // union for CRC check
    unsigned short i;                    // as an unsigned word
    unsigned char c[2];                  // as two characters
    struct bits{                         // as a bitfield
        unsigned msb:1;                  // most significant bit
        unsigned :14;
        unsigned lsb:1;                  // least significant bit
       }b;
    };

unsigned short CRC16(unsigned char *msg, int len)
{
    int i, j;
    CRC_doub sck, byt;
    unsigned short crc;

    sck.i=0xFFFF;                                  // set initial remainder

    for (i=0; i<len; i++)                          // for each byte in buffer
       {
        byt.c[0]=msg[i];                           // put the current character at end of working byt
        byt.c[1]=0;                                // set start of byt to 0
       sck.i^=byt.i;                               // set sck to sck XOR byt (MOD-2 maths)
        for (j=0; j<8; j++)                        // for each bit
           {
            if (sck.b.msb)
              {
                sck.i>>=1;                         // shift the remainder right 1 bit (divide by 2)
                sck.b.lsb=0;                       // and set the MSB to 0
               sck.i^=CRC16_POLYNOMIAL;            // set remainder = sck XOR the CRC16 polynomial
              }
            else
              {
                sck.i>>=1;                         // shift the remainder right 1 bit (divide by 2)
                sck.b.lsb=0;                       // and set the MSB to 0
              }
           }
       }

    crc = (unsigned short)((sck.i)<<8); // update the CRC in the data structure
    crc |= (unsigned short)((sck.i&0xFF00)>>8);

    return crc;
}
```


EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

Sample Decompression Code:

```
char *CompBuffer;                    //Compressed Data in DA2 segment
int BitIndex;                        //Bit Index into CompBuffer

u_int32_t GetSymbol(int BitsPerSymbol)
{
    int bp = BitIndex >> 3;      //Byte Index into CompBuffer
    int bi = BitIndex & 7;       //Bit Index into bp

    u_int32_t s = 0;
    unsigned char m = 0x80 >> bi;
    unsigned char b = CompBuffer[bp];
    for (int i=0; i<BitsPerSymbol; i++)
        {
        s <<= 1;
        if (b & m)
            s |= 1;

        if ((m >>= 1) == 0)
            {
            b = CompBuffer[++bp];
            m = 0x80;
            }
        }

    BitIndex += BitsPerSymbol;

    return s;
}

int32_t GetSample(int BitsPerSymbol)
{
    u_int32_t n;
    int32_t v = 0;
    int c = 0;

    int DataBitsPerSymbol = BitsPerSymbol - 1;
    int32_t tm = 1 << DataBitsPerSymbol;
    int32_t sb = tm >> 1;
    int32_t sm = tm - 1;

    do
        {
        n = GetSymbol(BitsPerSymbol);
        if (c == 0)
            {
            if (n & sb)
                v = -1;
            }
        v <<= DataBitsPerSymbol;
        v |= (n & sm);
        c = 1;
        }
    while (!(n & tm));

    return v;
}
```

```
int DecodeSamples(char bytesper,
                  char compression_info,
                  le_u_int16_t sample_rate
                  int32_t *samples)
{
    int32_t first, last;
   int BitsPerSymbol = (int)compression_info;

   if (BitsPerSymbol == 0)      //Raw Data!
      {
      unsigned char *p = CompBuffer;
       for (int i=0; i<sample_rate; i++)
         {
          int32_t sample_val = 0;
           switch(bytesper)
            {
             case 1:
                  sample_val = *(int8_t *)p;
                 break;
             case 2:
                  sample_val = p[1];
                 sample_val <<= 8;
                  sample_val |= p[0];
                 sample_val <<= 16;
                 sample_val >>= 16;
                  break;
             case 3:
                  sample_val = p[2];
                 sample_val <<= 8;
                  sample_val |= p[1];
                 sample_val <<= 8;
                  sample_val |= p[0];
                 sample_val <<= 8;
                 sample_val >>= 8;
                  break;
             case 4:
                  sample_val = *(le_int32_t *)p;
                  break;
            }
          samples[i] = sample_val;
          p += bytesper;
         }
       return 1;
      }

   first = *(le_int32_t *)&CompBuffer[0];
   last = *(le_int32_t *)&CompBuffer[4];

   int32_t cur = first;
   int32_t diff;
   BitIndex = 2 * sizeof(u_int32_t) * 8;
   samples[0] = first;
   for (int i=1; i<sample_rate; i++)
      {
       diff = GetSample(BitsPerSymbol);
       samples[i] = cur + diff;
       cur = samples[i];
      }

   return (cur == last);
}
```

5.4 Command Strings

All Commands begin with $ followed by at 2 chars that define the command. Some commands require further data bytes usually as two binary 8 bit values.

Command	Data Bytes	Function	Data Format
$OK	0	Confirm Baud Rate Change	
$BA	2	Set Baud Rate	Two bytes of data. 1st byte = 00 (NULL) 2nd byte in binary 03 = 9600 04 = 19200 05 = 38400 06 = 57600 07 = 115200 08 = 230400
$CA	2	Set Cal Mode	Two bytes of data 1st byte = 00 (NULL) 2nd byte in binary 00 = Off 05 = Auto 01 = +5volt 06 = +.3125v 02 = 0 volt 07 = -.3125v 03 = -5volt 08 = .3125 sine 04 = +5 volt Sine 09 = .3125 dual sine
$FT	2	Set Filter Type	00 = Linear Phase
$LI	2	Line Output Settings	Not used
$LP	0	Set Low Power Mode	Not used
$SP	2	Set PSU's	Not used
$SR	2	Set sample rate per block	Not used
$NB	2	Set No of bytes per sample	3 = 3 bytes 4 = 4 bytes
$SC	2	Component Settings	See chapter 4.7 for more information bit 0 = Enable Vert Component bit 1 = Enable N/S Component bit 2 = Enable E/W Component bit 3 = Set Vert gain high bit 4 = Set N/S gain high bit 5 = Set E/W gain high

Command	Data Bytes	Function	Data Format
$SF	2	Set Filter Ratio (Set Sample Rate)	Data is binary. first byte: 3 = filter no 3 4 = filter no 4 5 = filter no 5 6 = filter no 6 second byte: decimation ratio filter 3 = 1,2,3,4,6,or 12 filter 4 = 1,2,5,or 10
$RW	0	Write settings to ROM	
$RD	0	Restore Default ROM	
$RP	14	Buffered Data Retransmission	See section 5.5 for Data Format
$VL	2	Set Very Low Gain	See section 5.6 for more information
$VH	2	Set Very H Gain	See section 5.6 for more information
$SM	2	Set Satellite Mode	See section 5.7 for more information
$SI	2	Set Initial Position	See section 5.7 for more information

5.5 Data Retransmission

This section describes how to get the Earth Data digitizer to re transmit data that may have been missed or lost due to corruption or other reason.

Data is requested for re-transmission by the host issuing a request for a specific second that is missing or a set of seconds that are missing. The request form is a short ASCII packet containing the second identification and how many seconds to retransmit. A simple checksum is added to the packet to validate the request. If the request is accepted the requested data is inserted in the data stream after any current packet has been transmitted. When the retransmitted second data or set of data has been sent the digitizer resumes sending from where it left off.

The digitizer checks that the requested data is available in its buffer. Providing the data is available it will be inserted in the data stream immediately if it is free or following any data currently being sent.

If the request fails the checksum test or the data is not available the request is quietly discarded.

The digitizer retains between 30 minutes and 8 hours of data in its buffers (depending on sample rates and number of channels) before it is discarded. This means that at any time the host may be unavailable for an hour before data is lost.

The format of the request follows: -

$RPTTTTTTTTNNNNSS

The request is a string of 17 ASCII characters all in upper case. The values for TTTTTTTT, NNNN and SS are all hexadecimal representation in ASCII. Again the hexadecimal characters A to F must be in upper case.

$RP is the retransmit packet command.

TTTTTTTT is an eight-character hexadecimal value representing the second of the requested packet. This is a second count started on the 1st January 1970 as defined by the original IBM PC time standard. The time is represented by a 32 bit unsigned integer count which started at 0 on Jan 1st 1970 at 0000 UTC. Convert this count to hexadecimal and send eight ASCII characters.

NNNN is a four-character count defining how many second packets are to be retransmitted. A value of "0001" will cause a retransmit of exactly one seconds worth of data, "0002" two seconds etc. A value of "0000" is special and causes the transmission to be repositioned to the time specified by TTTTTTTT. Transmission of current data will cease and transmission will resume at the time specified by TTTTTTTT. The digitizer will send data from this time until current data is reached. This count is represented as four hexadecimal ASCII characters. For example if 35 seconds are required, convert to hexadecimal to obtain 0x23, then send "0023" as four ASCII characters.

SS is a two-character hexadecimal sum check of the previous 15 characters.
Add the preceding 15 characters together to obtain a checksum. Reduce it to the least significant 8 bits. The checksum "SS" is the hexadecimal representation of the two four bit nibbles in the byte send as the high nibble first.

The request must be exactly seventeen characters long. Leading zeros in the count field must not be suppressed.

There are two ways of calculating this time count.

Firstly it can be derived from the GPS string in the data Header (in the MOD section). Earth Data can provide code in "C" to generate the time count if required.

Secondly and possibly an easier method is to use the second count planted in the data header just for this purpose. This can be used to verify contiguous data is being received and provide the second identification of any missing packets.
The second count is stored in the header in as a hexadecimal integer.
The four-byte field is contained in the MOD section and starts at offset 103 (decimal) from the start of the section.

Ordered as bytes, offset 103 is the least significant byte with offset 106 the most significant byte. Convert this four-byte field to its eight-character ASCII hexadecimal equivalent and you have the time code that is TTTTTTTT.

5.6 Channel Selection & Gains

Selecting channels and their gains is performed with a combination of two commands. The Set Components command $SC selects which channels are active and which of normal (low) gain or high gain is selected. A second Very Low Gain $VL overrides the low/high gain setting with a very low gain setting. The $VL command is only supported by issue 4 digitizers and later. It should not be used with earlier units. When the $VL command is used send the $SC command first followed by the $VL command. Six channels require two $VL commands, one for each set of three channels.

The $SC command has the form

$SC<D1><D2> where D1 & D2 are two binary data bytes.

Bit Position	D1	D2
0	1 = channel 4 on	1 = channel 1 on
1	1 = channel 5 on	1 = channel 2 on
2	1 = channel 6 on	1 = channel 3 on
3	1 = channel 4 Hi gain	1 = channel 1 Hi gain
4	1 = channel 5 Hi gain	1 = channel 2 Hi gain
5	1 = channel 6 Hi gain	1 = channel 3 Hi gain
6	0	0
7	0	0

The $VL command has the form

$VL<D1><D2> where D1 & D2 are two binary data bytes

Bit Position	D1	D2
0	0 = Select channels 1..3 1 = Select channels 4..6	0 = Disable Very Low gain 1 = Enable Very Low gain
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0

The $VH command has the form

$VH<D1><D2> where D1 & D2 are two binary data bytes

Bit Position	D1	D2
0	0 = Select channels 1..3 1 = Select channels 4..6	0 = Disable Very High gain 1 = Enable Very High gain
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0

5.7 GPS Commands

Two commands are available to aid a Trimble GPS in getting initial lock.

$SM sets the satellite mode.

$SM00 (the default) sets multiple satellite mode and is the normal mode the GPS operates in. The GPS needs a good view from at least three satellites to obtain good GPS lock.

$SM01 sets single satellite mode which allows GPS lock from a single satellite with potentially less accurate time & position accuracy.

Note: 00 and 01 are two binary bytes.

$SI<datastring> passes an ASCII data string to a Trimble GPS. The usual use for this command is to set the GPS initial position by passing world coordinates to the GPS. The general format of the data string is >SIP longitude latitude altitude<

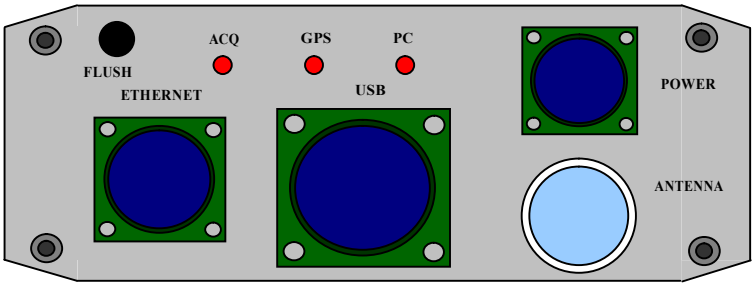
If a GPS knows its approximate position prior to first use, the time to the first lock may be reduced considerably.

The data string to set a southern England location is >SIP+50-001+0010<

Refer to the Trimble GPS manual for a full description of these two commands.

6 Connectors

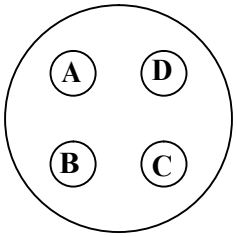
6.1 Output Panel



6.1.1 Power Connector (4 way)

Pin	Panel plug. Part No. MIL-C-26482 08-04P
A	Power in (+12volt nominal)
B	
C	
D	GND

Mating connector Part No. MIL-C-26482 08-04S



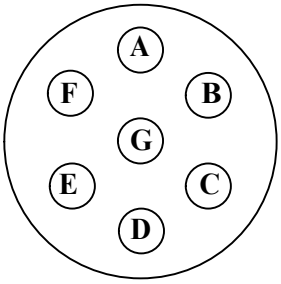
View looking into plug on digitizer endplate.

EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

6.1.2 Ethernet/Modmon RS485 Connector (7 way)

Pin	Panel plug. Part No. MIL-C-26482 10-07P
A	ETHERNET-ETX+
B	ETHERNET-ETX-
C	ETHERNET-ERX+
D	ETHERNET-ERX-
E	RS485-RX
F	RS485-TX
G	RS485-GND

Mating connector Part No. MIL-C-26482 10-07S



6.1.3 GPS Antenna Connector

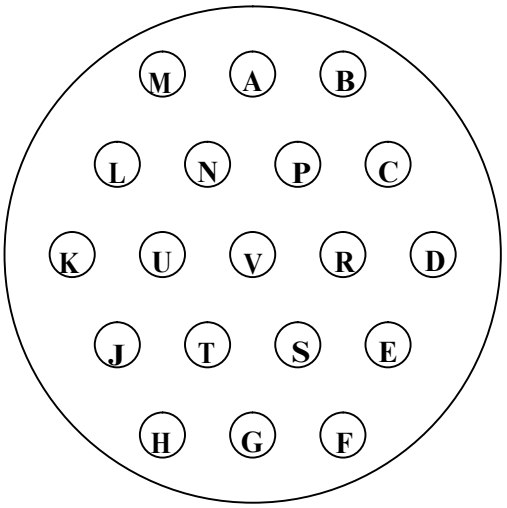
The GPS connector is a standard sealed TNC.

6.1.4 Multi Connector (19 way)

Pin	Panel plug. Part No. MIL-C-26482 14-19P
A	USB-DMI
B	USB-DPI
C	USB-GND
D	
E	MODMON 422/485 R+/A
F	MODMON 422/485 T-/B
G	MODMON 422/485 T+/A
H	MODMON 422/485 GND
J	
K	CONSOLE-GND
L	CONSOLE-RXD *
M	USB-POWER (5v)
N	-DEFIN
P	
R	
S	MODMON 422/485 R-/B
T	MODMON422/485 GND
U	CONSOLE-TXD **
V	

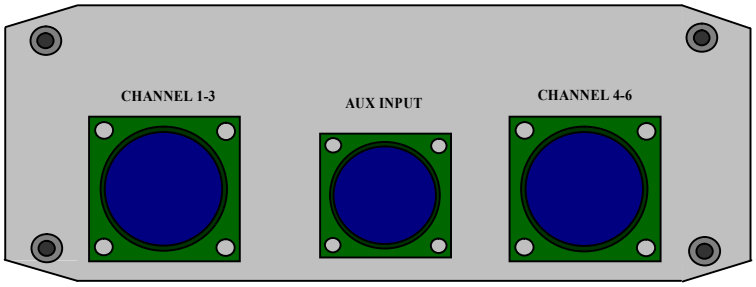
* RXD = Receive by EDR210, ** TXD = Transmit from EDR209

Mating connector Part No. MIL-C-26482 14-19S



EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

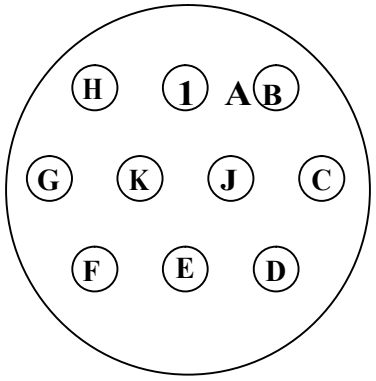
6.2 Input Panel



6.2.1 Sensor Connectors (10 way)

Pin	Panel sockets. Part No. MIL-C-26482 12-10S
A	Input1+ or Input4+
B	Input1- or Input4-
C	Input2+ or Input5+
D	Input2- or Input5-
E	Input3+ or Input6+
F	Input3- or Input6-
G	CAL OUT
H	0V
J	- 24V
K	+24V

Mating connector Part No. MIL-C-26482 12-10P

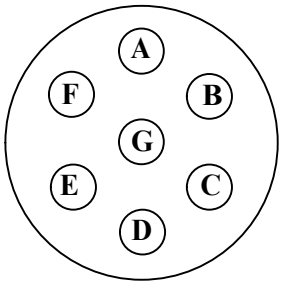


EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

6.2.2 AUX Input Connector (7 way)

Pin	Panel socket. Part No. MIL-C-26482 10-07S
A	Low-res Analogue In1
B	Low-res Analogue In2
C	Low-res Analogue In3
D	GND
E	Digital out (open collector)
F	Digital out (TTL)
G	Low-res Analogue In4

Mating connector Part No. MIL-C-26482 10-07P



EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

7 Specification

7.1 Input

Number:	3 or 6 (optional)
Type:	Differential
Input impedance:	1Mohm ±5% to gnd 2Mohms differential
Input CMRR:	Better than 100dB
Input Sensitivity:	
Input Range:	2.5µV per bit: 100nV per bit (RS232 switchable)
Gain accuracy:	± 0.8 Volt, ± 8 Volt and ± 20 Volt (User Selectable)
d.c. offset:	± 0.1dB after auto cal ± 5uV low gain (after autocal) ± 1uV high gain (after autocal)
offset voltage drift:	± 10uV/°C
crosstalk:	Better than -100 dB, any frequency within passband

7.2 A to D

Number:	1 to 6 (User selectable)
Type:	Earth Data 7th order Sigma Delta
Sampling frequency:	192kHz
Filter type:	cascaded FIR
Output sampling ratio:	2.5:1 (All sampling frequencies)
Filter response:	Linear phase.
Alias protection:	±0.1 dB d.c. to 0.4 sample rate Spurious in band signals better than -120dB's below full scale output as a result of any frequency above the passband and any amplitude up to full scale.
Output sample rates:	1,2,3,4,10,20,25,40,50,75,100,150,200,250,300 375,500,600,750,1000,3000
rms to rms noise:	140 dB @ 100 samples/ second 130 dB @ 1000 samples/second
Spurious Tones:	Nothing above -144 dB's (Input short circuit)
Intermodulation Products:	Less than -120 dB's for any two sine wave frequencies within the passband at half full scale amplitude.

7.3 Additional Connector Functions

Sensor power:	12 volt ± 0.2 volt supply available for driving active sensors. 100mA capability

7.4 Additional A to D Channels

Number:	3 or 4
Type:	12 bit Successive approximation.
Input Impedance:	10K
Input range:	±10volts
Sampling rate:	1 sample/second

7.5 Communication

Output format:	RS232 bi-directional RS485 half duplex
Input format:	RS232
RS232 baud rates:	9600,19200,38400,57600,115200,230400
RS485 baud rates:	As above
10-BaseT	Ethernet
100-BaseT	Ethernet

7.6 Additional Features

Serial Number:	All units transmit a unique serial number within the RS232 data.
System monitoring:	Supply current, supply volts and internal temperature reported in RS232 data.
GPS	Earth Data GPS unit synchronises digitizer to UTC
Auto Cal:	Gain and offset calibration with RS232 commands.

7.7 Environmental

Housing:	Extruded Aluminium
Protection:	IP67
Operating Temperature:	
Storage Temperature:	- 20 to + 50 °C
Relative Humidity:	- 40 to + 125 °C 95 %

7.8 Power Requirements

Supply voltage:	12 to 28 volts d.c.
Supply Current:	120mA (3ch.) or 160mA (6ch.) at 12 volts.

7.9 Physical

Size:	175mm x 108mm x 45mm
Weight:	1.1 kg

8 Accessories

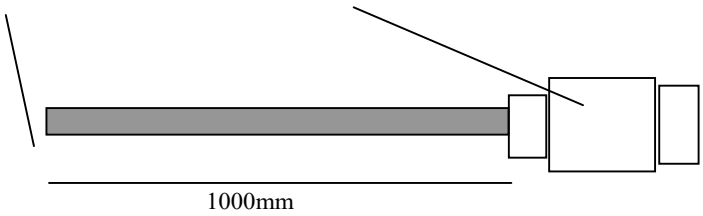
8.1 Connecting Cables

Free ends for termination by user

8.1.1 Sensor Cable

(Order Code WA2315)

10 way cable	**to**	**Free plug. (10 way)** *(Part No. MIL-C-26482 12-10P)*			
red	**to**	**A**	*INPUT1+*	*or*	*INPUT4+*
blu	**to**	**B**	*INPUT1-*	*or*	*INPUT4-*
grn	**to**	**C**	*INPUT2+*	*or*	*INPUT5+*
yel	**to**	**D**	*INPUT2-*	*or*	*INPUT5-*
whi	**to**	**E**	*INPUT3+*	*or*	*INPUT6+*
blk	**to**	**F**	*INPUT3-*	*or*	*INPUT6-*
brn	**to**	**G**	*CALOUT*		
vio	**to**	**H**	*GND*		
ora	**to**	**J**	*GND (Optional -12 V POWER OUT)*		
pnk	**to**	**K**	*+12 V POWER OUT*		

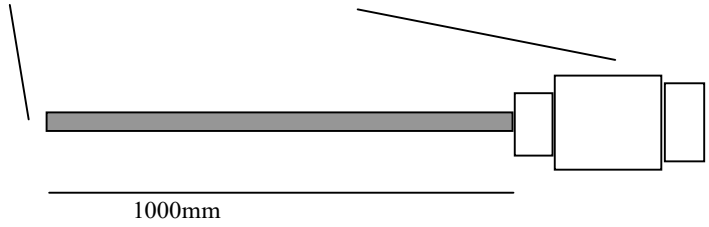


The screen is not connected in the plug, but should be connected at the remote end.

8.1.2 Auxiliary Cable

(Order Code WA2316)

10 way cable	to	Free Plug (7way) (*Part No. MIL-C-26482 10-07P)*	
red	**to**	**A**	*LOW RES ANALOG IN 1*
blu	**to**	**B**	*LOW RES ANALOG IN 2*
grn	**to**	**C**	*LOW RES ANALOG IN 3*
yel	**to**	**D**	*GROUND*
whi	**to**	**E**	*DIGOUT 1 (Open collector)*
blk	**to**	**F**	*DIGOUT 2 (TTL)*
brn	**to**	**G**	*DIGIN 1/LOW RES ANALOG IN 4*

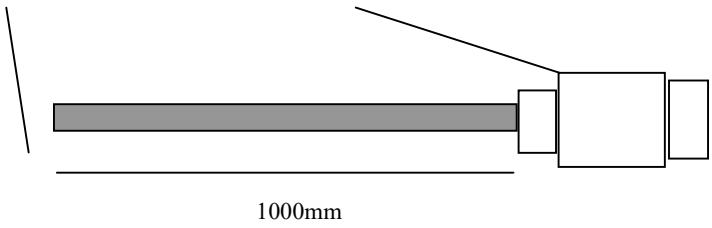


The screen is not connected in the plug, but should be connected at the remote end.

8.1.3 Serial Cable

(Order Code WA2317)

10 way cable	to	Free socket (7 way) *(Part No. MIL-C-26482 10-7S)*	
whi	**to**	**E**	*MONRX*
blk	**to**	**F**	*MONTX*
brn	**to**	**G**	*MONGnd*

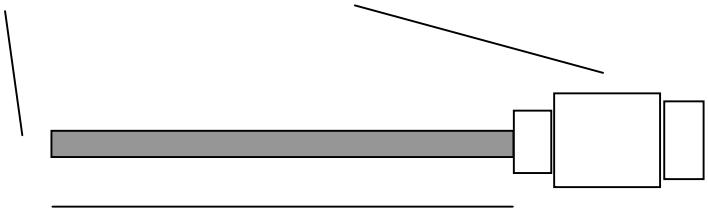


EARTH DATA
A MEMBER OF THE KENDA GROUP
scientific recording

8.1.4 Power Connector

(Order Code WA2318)

4 way cable	**to**	**Free socket** *(Part No. MIL-C-26482 08-04S)*	
brn	**to**	**A**	*POWER IN*
blu	**to**	**D**	*GND*



1000mm

8.2 Connecting Cables

Terminated both ends.

1) **GPS extension cable. (Order Code WA2331)**
2) **RS232 (MIL-C TO 9 WAY D-TYPE) (Order Code WA2320)**

8.3 Connectors

1) **SENSOR CONNECTOR (Order Code CN/443/12-10-P)**
2) **AUXILIARY CONNECTOR (Order Code CN/443/12-7-P)**
3) **POWER CONNECTOR (Order Code CN/443/8-4-S)**
4) **SERIAL CONNECTOR (Order Code CN/443/10-7-S)**
5) **USB/EXT. GPS CONNECTOR (Order Code CN/443/14-19-S)**

9 Upgrade Procedure

From time to time a new firmware issue will be available. The issuing of new software will be communicated to customers by e-mail or on the company download site. A password will be required for access to the download area.

http://www.earthdata.co.uk/down.html

The software upgrade procedure is described in separate documentation.

10 Warranty and Returns

All Earth Data products are warranted against defective materials and workmanship for a period of one year from the date of delivery to the original purchaser. Any product reported as being defective within the warranty period will be either repaired or replaced at the discretion of Earth Data This warranty does not apply to products that have been damaged by improper use.

In the event of an equipment failure, it is necessary to notify Earth Data immediately and obtain a Return Authorisation Number (RAN). Equipment returned without a RAN will not be repaired until notification is received from the purchaser and the RAN issued. The transportation costs for return are the responsibility of the customer. The date of issuing the RAN will be used for determining warranty status. Earth Data will pay transportation costs back to the customer

If the equipment is outside the warranty period then the purchaser must supply a purchase order number before repairs can proceed. Most Earth Data equipment carries a standard investigation fee, which will be communicated to the purchaser at the time of the return or on request.

CE

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP